

VŠB – Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra kybernetiky a biomedicínského inženýrství

# **Automatizace procesu výroby integrací manipulačního mechanismu**

## **Automation of the Production Process by Integration of the Handling Mechanism**

# Zadání bakalářské práce

Student:

**Petr Janoušek**

Studijní program:

B0714A150001 Řídicí a informační systémy

Téma:

**Automatizace procesu výroby integrací manipulačního mechanismu**

**Automation of the Production Process by Integration  
of the Handling Mechanism**

Jazyk vypracování:

čeština

Zásady pro vypracování:

Cílem bakalářské práce je analýza, návrh a realizace technologie pro začlenění manipulačního mechanismu do výrobní linky. Původní systém je složen z průmyslových komponent a řízení, které nejsou připraveny pro automatizovanou výrobu.

V souhrnu je práce charakterizována následujícími body:

1. Analýza řízení manipulačního mechanismu.
2. Zhotovení technologické dokumentace současného řešení.
3. Integrace mikrokontrolérem řízeného manipulačního stroje do výrobní linky.
4. Návrh vizualizace HMI pro výrobní linku.
5. Realizace a test systému s ohledem na vybrané prvky Industry 4.
6. Zhodnocení výsledků řešení.

Seznam doporučené odborné literatury:

- [1] VODA, Zbyšek. *Průvodce světem Arduina*. Praha: BEN - technická literatura, 2017. ISBN 978-80-87106-90-7.
- [2] PUŽMANOVÁ, Rita. *Moderní komunikační sítě od A do Z*. Brno: Computer Press, 2006. ISBN 978-8025112786.
- [3] MONK, Simon. *Programming the Raspberry Pi: Getting Started with Python*. McGraw-Hill Education TAB, 2012. ISBN-13: 978-0071807838.
- [4] HOROWITZ, Paul a Winfield HILL. *The Art of Electronics*. Cambridge University Press, 2015. ISBN-13: 978-0521809269.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Zdeněk Slanina, Ph.D.**

Datum zadání: 01.09.2019

Datum odevzdání: 30.04.2020



---

doc. Ing. Jiří Koziorek, Ph.D.  
*vedoucí katedry*

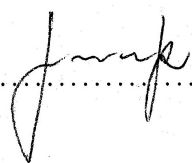


---

prof. Ing. Pavel Brandštetter, CSc.  
*děkan fakulty*

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 15. května 2020

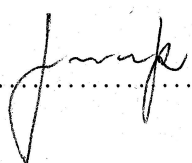


.....



Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava.

V Ostravě 15. května 2020



.....

Rád bych zde poděkoval svému tátovi za jeho trpělivost a úsilí, kterým mi umožnil tuto práci realizovat. Rád bych také poděkoval svému vedoucímu Ing. Zdeňku Slaninovi, Ph.D., za jeho rady, podporu a občasné povzbuzení. A nakonec, rád bych poděkoval inženýrům Janu Dedkovi, Matěji Golembiovskému a Tomáši Dočekalovi, za to že si vždy najdou chvíli, dobře mi poradit.

## **Abstrakt**

Cílem této práce je vytvoření robotického manipulátoru, který svou činností automatizuje výrobu tak, že bude synchronizovaně obsluhovat dílčí automatická pracoviště výroby. Zcela klíčovým požadavkem na tento stroj je minimalizace pořizovacích nákladů, což je podmínka praktické realizace. K již existující mechanické konstrukci jsou v této práci navrženy potřebné pohony, snímače, elektronika rozvaděče a programová výbava. Automatizace pomocí manipulátoru je zajištěna přidáním centrální řídicí jednotky, úpravou programů jednotlivých strojů a jejich spojení komunikační sběrnici. Uživatelské ovládání celé výroby je zajištěno jednotným uživatelským rozhraním, které bylo naprogramováno v několika verzích pomocí různých programovacích jazyků. Pro řešené části (rozvaděč, programy), je zhotovena technická dokumentace, jejíž část týkající se manipulátoru je k práci přiložena. Navržené řešení je v závěrečné části realizováno a otestováno v provozu.

**Klíčová slova:** Manipulátor; Automatická výroba; Průmyslová integrace; HMI; ATmega2560; Bakalářská práce

## **Abstract**

The main goal of this bachelor thesis is to create a robotic manipulator, which would automate a specified manufacturing process by synchronously controlling and operating multiple individual machines. In its design the decisive requirement is a minimalization of its initial cost, required for practical realisation. Motors, sensors, distribution board and software is added to a mechanical construction, which was already predesigned. The automation is achieved by designing and adding central control unit, redesigning programs of the individual machines and connecting them with a suitable communication interface. User interface software was designed in multiple versions using different programming tools. Finally, for practical purposes, a technical documentation was made for the whole solution, and the sections that are related to the manipulator are included. Designed solution is constructed and tested in operation.

**Keywords:** Robotic manipulator, Automatic production, Industrial integration, HMI, ATmega2560, Bachelor thesis

# Obsah

<b>Seznam použitých zkratk a symbolů</b>	<b>10</b>
<b>Seznam obrázků</b>	<b>11</b>
<b>Seznam tabulek</b>	<b>12</b>
<b>Seznam výpisů zdrojového kódu</b>	<b>13</b>
<b>1 Úvod</b>	<b>14</b>
1.1 Specifikace úlohy a postupu řešení . . . . .	15
<b>2 Potřebné pojmy robotiky</b>	<b>16</b>
2.1 Robot a manipulátor . . . . .	16
2.2 Mechanika a kinematika . . . . .	17
2.3 Struktura a členění . . . . .	18
<b>3 Prostředky řešení</b>	<b>20</b>
3.1 Řídicí subsystém . . . . .	20
3.2 Vnímací subsystém . . . . .	24
3.3 Akční subsystém . . . . .	26
3.4 Koncový efektor . . . . .	27
<b>4 Integrace do výrobního procesu</b>	<b>28</b>
4.1 Popis výrobního procesu . . . . .	28
4.2 Ohradová paleta . . . . .	29
4.3 Výchozí vybavení strojů . . . . .	30
4.4 Návržení vhodných úprav . . . . .	30
<b>5 Realizace manipulátoru</b>	<b>31</b>
5.1 Popis mechaniky . . . . .	31
5.2 Pohony . . . . .	32
5.3 Senzory . . . . .	33
5.4 Rušení . . . . .	34
5.5 Rozvaděč . . . . .	36
5.6 Řídicí jednotka manipulátoru . . . . .	37
5.7 Hlavní řídicí jednotka . . . . .	38
5.8 Programové vybavení . . . . .	38
5.9 Komunikace . . . . .	39
5.10 Datové rámce . . . . .	40

<b>6</b>	<b>Postup sestavení</b>	<b>42</b>
<b>7</b>	<b>Dokumentace</b>	<b>44</b>
7.1	Eplan . . . . .	44
7.2	Zapojení rozvaděče . . . . .	44
<b>8</b>	<b>Uživatelské rozhraní</b>	<b>45</b>
8.1	Hardwarové prostředky . . . . .	45
8.2	Softwarové prostředky . . . . .	46
8.3	Návrh rozhraní . . . . .	46
8.4	Rozhraní v C# . . . . .	47
8.5	Rozhraní v LabView . . . . .	49
<b>9</b>	<b>Sběr dat a testování</b>	<b>50</b>
9.1	Testování . . . . .	50
9.2	Ukládání dat . . . . .	50
<b>10</b>	<b>Závěr</b>	<b>53</b>
	<b>Literatura</b>	<b>54</b>
	<b>Přílohy</b>	<b>55</b>
<b>A</b>	<b>Dokumentace DPS</b>	<b>56</b>
<b>B</b>	<b>Dokumentace Eplan</b>	<b>59</b>
<b>C</b>	<b>Uživatelská rozhraní</b>	<b>64</b>

## Seznam použitých zkratek a symbolů

CAN	–	Controlled Area Network (Průmyslová sběrnice)
CNC	–	Computer Numeric Control (Číslicové řízení)
ČSN	–	Computer Numeric Control (Číslicové řízení)
DIN	–	Deutsches Institut für Normung (Německý ústav pro normalizaci)
DIP	–	Dual Inline Package (Pouzdro se dvěma řadami pinů)
DPS	–	Deska plošných spojů (Printed circuit board)
DoF	–	Degree of Freedom (Stupeň volnosti)
EMC	–	ElectroMagnetic Compatibility Plánování podnikových zdrojů)
EMI	–	ElectroMagnetic Interference
ERP	–	Enterprise Resource Planning (Plánování podnikových zdrojů)
GTK	–	Gimp ToolKit (Knihovna pro tvorbu GUI)
GUI	–	Graphical user interface (Grafické uživatelské rozhraní)
HTML	–	HyperText Markup Language (Hypertextový značkový jazyk)
HMI	–	Human Machine Interface (Rozhraní člověk-stroj)
IPC	–	Industrial Personal Computer (Průmyslový počítač)
ISO	–	Industrial Personal Computer (Průmyslový počítač)
KP	–	Kinematic Pair (Kinematická dvojice)
LV	–	LabView (Vývojové prostředí)
MCU	–	MicroController Unit (Mikrokontrolér)
PC	–	Personal Computer (Osobní počítač)
PLC	–	Programmable Logic Controller (Programovatelný logický automat)
RTOS	–	Real-Time Operating System (Operační systém reálného času)
RT	–	Real-Time (V reálném čase)
SCADA	–	Supervisory Control And Data Aquisition (Dispečerské řízení a sběr dat)
SMD	–	Supervisory Control And Data Aquisition (Dispečerské řízení a sběr dat)
USB	–	Universal Serial Bus (Seriová sběrnice)
VI	–	Virtual Instrument (Virtuální přístroj)
VS	–	Vestavěný Systém

## Seznam obrázků

1	Znázornění translace a rotace tělesa v prostoru a rovině . . . . .	17
2	Schématické znázornění kinematické dvojice posuvné a rotační . . . . .	17
3	Srovnání paralelního [4] a seriového robotu . . . . .	18
4	Blokové schéma robotického manipulátoru . . . . .	19
5	Blokové schéma měřicího řetězce . . . . .	26
6	Malá ohradová paleta a její části . . . . .	29
7	Kinematická struktura PPP manipulátoru . . . . .	31
8	Koncový efektor se 4 elektromagnety [12] . . . . .	32
9	DPS s obvodem pro čtení I/O vstupů . . . . .	35
10	Zapojení rozvaděče manipulátoru . . . . .	36
11	Umístění ovládacího rozhraní z pohledu komunikace . . . . .	40
12	Rozvaděč v průběhu montáže . . . . .	42
13	Zobrazení různých intenzit rušení na osciloskopu . . . . .	43
14	Ovladač hydraulického lisu . . . . .	45
15	Staré rozhraní v C# . . . . .	47
16	Nové rozhraní v C# . . . . .	48
17	Nové rozhraní v LabView . . . . .	49
18	Rohové díly na odkládací paletě . . . . .	51

## Seznam tabulek

1	Parametry různých typů palet MARS . . . . .	29
---	---	----



## Seznam výpisů zdrojového kódu

1	Ukázka G-kódu, obrys vnitřní rohové vzpěry . . . . .	23
2	Ukázka funkcí pro zasílání dat po sériové lince . . . . .	41

# 1 Úvod

Už od časů průmyslové revoluce, tedy přibližně konce 18. století, se spolu se zrychlujícím se rozvojem techniky šíří mezi lidmi strach z nahrazování lidské práce stroji. Tyto myšlenky vedly ke vzniku luddistických hnutí rozbíječů strojů, většinou příslušníků dělnické třídy, kteří věřili, že mechanizace a automatizace výrobních procesů je připraví o práci a uvrhne do hmotné nouze. Již v roce 1819 byl v Brně jízdním vojskem rozeznán dav přibližně 150 dělníků, protestujících proti zavedení postříhačského stroje ve vlnářské manufaktuře. [1]

Zavedení automatizace je však také typicky velice nákladný proces, pro mnohé malé firmy prakticky nerealizovatelný. Každé větší rozšíření sortimentu produktů vede na další podobně rozsáhlé investice spojené s pořízením nových specializovaných strojů. Zároveň většina strojů stejně jako jejich součásti podléhá určitému životnímu cyklu, což vede často k tomu, že je stroje zejména v těžším průmyslu každých několik let vyměnit za zcela nové. Po zvážení těchto nákladů může být pro malé a začínající firmy taková investice velice riziková nebo přímo nerealizovatelná.

Dva výše uvedené základní problémy daly podnět ke vzniku zadání této práce. Možným řešením je snížit nároky na kvalitativní parametry takového stroje a tím minimalizovat náklady na jeho pořízení. Takový stroj pak může markantně urychlit vývoj drobné firmy. Ta totiž v rozporu s výše uvedenými luddistickými myšlenkami může výhodně rozšířit své výrobní kapacity tak, že části své činnosti automatizuje, aniž by propustila své stávající zaměstnance. Ty může buď přesunout na pozice, které vyžadují lidský dozor, nebo je rekvalifikovat na pozice nově vzniklé zavedením automatizace.

Tato práce navazuje na snahu automatizovat výrobu v malé firmě, zabývající se recyklací a výrobou průmyslových obalů. V rámci zmíněného procesu již byly navrženy a vyrobeny stroje schopné vyrábět všechny potřebné díly. Pro jejich činnost je však potřeba lidská obsluha, což je časově i finančně neefektivní. Dalším krokem tohoto úsilí bude tedy spojení všech těchto strojů do celku schopného automatické výroby. To bude umožněno přidáním prvku spojující dílčí kroky výroby, schopném přenášet obrobky mezi jednotlivými pracovišti a výrobu tak synchronizovat.

Výše zmíněným zařízením bude robotický manipulátor, jehož mechanická část byla již navržena i sestrojena. Tato práce si tedy klade za cíl vybavit robot potřebnou elektronikou a programy tak, aby byl plně schopný funkce.

Specifickým požadavkem, vzhledem k výše zmíněnému je udržení nákladů na tento manipulátor na naprostém minimu. Cílem této práce je tedy optimalizace nároků a vybavení, stejně tak jako realizace stroje tak, aby na těch nejnižších možných nákladech byl schopen spolehlivé práce.

## 1.1 Specifikace úlohy a postupu řešení

Předchozí text naznačuje motivaci a cíle této práce. Toto téma je svým obsahem poměrně rozsáhlé, jelikož zahrnuje řadu oborů a podúloh, které je potřeba vyhodnotit a zpracovat. Zdá se nezbytné práci rozvrhnout a rozčlenit do níže popsanych částí. Nejdříve však bude vhodné, blíže rozvést přesné požadavky zadání a nastínit možnosti, jimiž by mohlo být těmto požadavkům vyhověno.

### Specifikace úlohy

Jak již bylo zmíněno výše, ve výchozím bodě této práce, je k dispozici mechanická konstrukce robotu. Tu bude třeba vybavit vhodnými pohony, snímači, elektronikou a programem. Větší část těchto zařízení bude umístěna v rozvaděči a ostatní budou namontovány na předem připravená místa.

Zejména pro usnadnění zapojení rozvaděče bude vytvořena příslušná technická dokumentace. Ta bude uvedena v příloze B. Dokumentace bude následně předložena ke kontrole jak vedoucímu práce, tak příslušným odborníkům a podle jejich připomínek upravena tak, aby bylo možné vystavit pro stroj prohlášení o shodě a uvést jej do provozu. Dokumentace v této práci je uvedena ve stavu, před úpravami navrženými těmito odborníky, čili podle návrhu autora práce.

To je však pouze část celého řešení, poněvadž tento robot má sloužit jako propojovací článek výrobní linky. Dílčí stroje této linky už jsou také hotovy a v provozu, bude však třeba upravit je tak, aby mohly synchronizovaně pracovat. K tomuto účelu bude přidána vhodná sběrnice a také bude potřeba vyřešit časování a synchronizaci úkolů.

Poslední částí této práce je návrh uživatelského rozhraní, vhodného pro ovládání jak celého procesu, tak manipulátoru samostatně. Toto rozhraní by mělo zejména umožňovat ovládání a indikovat základní stavy, případně sbírat základní informace pro případnou statistiku.

### Rozvržení práce

V kapitole 2.1 a jejích podkapitolách, jsou vymezeny některé základní pojmy robotiky používané dále v práci. Jednotlivé části kapitoly 3 se zabývají srovnáním možných použitelných komponent. Kapitola 4 pak uvádí informace, které bylo potřeba vzít v úvahu pro integraci manipulátoru do linky. V praktické části začínající kapitolou 5, je za použití dříve vybraných komponent sestaveno cílové řešení. Kapitola 7 uvádí bližší informace o tvorbě dokumentace. Následující kapitola 8 se věnuje tvorbě uživatelského rozhraní a nakonec v kapitole 9 je výsledný manipulátor vyzkoušen v testovacím provozu.

## 2 Potřebné pojmy robotiky

Robotika je poměrně moderní vědou, která v důsledku rychlého vývoje zahrnuje čím dále tím větší množství pojmů a podoblastí. Proto jsou níže vymezeny samotné pojmy robot a manipulátor tak, jak jsou užívány v této práci.

### 2.1 Robot a manipulátor

*Průmyslový robot může být definován jako programovatelné mechanické zařízení, používané namísto člověka, využívané pro vykonávání nebezpečných nebo repetitivních úloh, s vysokým stupněm přesnosti.* [2]

Robotika je moderním oborem se stále se rozšiřujícími oblastmi využití a nespočtem publikací, které se jí zabývají. Proto se nezdá nutné opakovat její historii, původ slova robot a tak podobně. Zdá se vhodnější zmínit, že předměty zájmu tohoto oboru - roboty, je možné velmi hrubě rozdělit do tří základních skupin.

- „Inteligentní“ a humanoidní roboti. Velmi sofistikovaní roboti, často implementující adaptivní algoritmy a strojové učení.
- Mobilní roboty, uzpůsobené pro pohyb v určitém prostředí a to buď autonomní nebo manuálně řízené.
- Manipulátory. Ve smyslu této práce myslíme roboty, jejichž účelem je zpřesňovat nebo zesilovat lidskou práci.

Následující práce se zabývá posledním z výše uvedených, proto je vhodné blíže specifikovat samotný pojem manipulátor. Ten se používá buďto ve významu stroje - robotu s určitými vlastnostmi, v jakém je použit ve výčtu uvedeném výše a nebo pro popis akčního ústrojí, čili mechanické části robotu. Přímo dle normy ISO, pak můžeme uvažovat tuto definici:

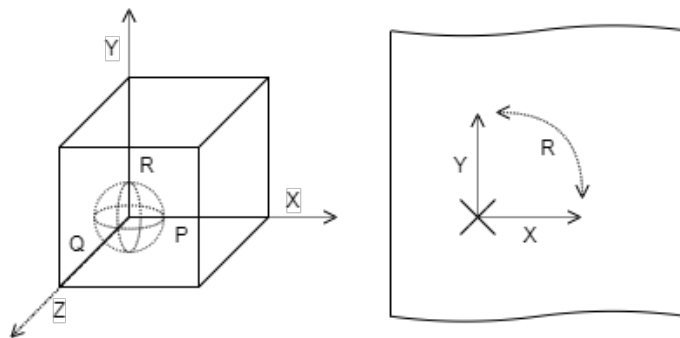
*Manipulátor je stroj, ve kterém mechanismus obvykle sestává ze sady členů (segmentů) navzájem spojených otočnou nebo posuvnou vazbou za účelem uchopení a/-nebo přemisťování předmětů (obrobků nebo nástrojů), obvykle v několika stupních volnosti.* [3]

Podle výše uvedených definic tedy můžeme zařízení, které je předmětem této práce nazývat jak slovem robot, tak slovem manipulátor, a to buď ve smyslu zařízení přemísťující různé předměty, nebo ve smyslu skupiny robotů. Slovo robot je navíc možné skloňovat jako životné či neživotné. Přisoudíme-li životný tvar „roboti“ humanoidním konstrukcím, zdá se logické průmyslová manipulační zařízení nazývat neživotným tvarem „roboty“.

## 2.2 Mechanika a kinematika

Pro potřeby praktické i další teoretické části je nejprve potřeba popsat sadu základních pojmů užívaných v robotice.

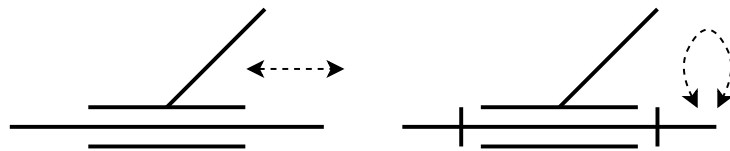
Prvním často užívaným pojmem je **stupeň volnosti**; zkratka DoF. Je jím popisován konečný počet parametrů, který zcela určuje pozici hmotného bodu nebo tělesa v prostoru. Jak názorně ukazuje obrázek 1. Bod v rovině je zcela určen dvěma stupni volnosti - dvěma souřadnicemi v této rovině. Těleso v rovině navíc může být o libovolný úhel rotováno - disponuje tedy třemi stupni volnosti. Stejně tak bod v prostoru je určen třemi stupni volnosti a těleso pak šesti, protože může být ve třech osách posouváno ale zároveň rotováno.



Obrázek 1: Znázornění translace a rotace tělesa v prostoru a rovině

Obecný robot sestává z pevných částí nazývaných **rameno**, které jsou navzájem spojeny **kloubem**. Kloub může být konstruován různými způsoby tak, aby umožňoval, určitý stupeň volnosti. V robotice se nejčastěji používá rotačních a posuvných kloubů, které přidávají jeden stupeň volnosti a znemožňují pohyb jakýmkoli jiným způsobem. Složitějšími typy kloubů jsou cylindrický a univerzální, které mají 2 DoF nebo sférický, který má 3. Spojení dvou ramen kloubem nazýváme **kinematická dvojice**.

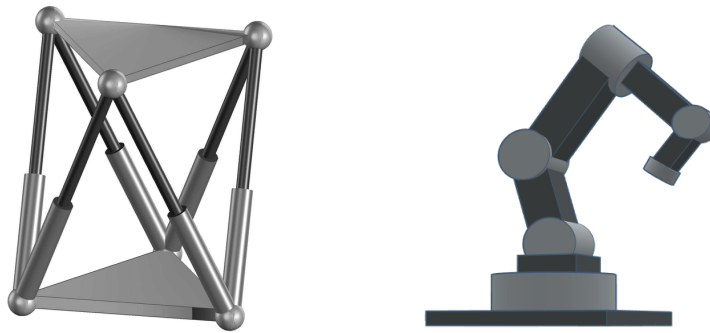
Výhodou kinematické dvojice posuvné (P-typ), a rotační (R-typ) je pak fakt, že je možné jednoduše je ovládat pomocí běžných aktuátorů. Umožnit řízený pohyb například sférického koloubu by bylo značně technologicky náročné a v praxi nákladné. Proto každý kloub (resp. kinematická dvojice), který není volný (tedy neovládaný), je v praxi téměř vždy řešen P nebo R kinematickou dvojicí, jejichž schematický náčrt je na obrázku 2.



Obrázek 2: Schématické znázornění kinematické dvojice posuvné a rotační

Jelikož robot se typicky skládá z více než jedné kinematické dvojice, užívá se také pojem **kinematická struktura** nebo kinematický řetězec. Ten popisuje množinu všech kinematických dvojic. K reprezentaci takové struktury se využívají grafy. Pokud jde o graf acyklický, tedy takový, který neobsahuje žádné smyčky, je kinematický řetězec otevřený viz obrázek 3 vpravo, v opačném případě se jedná o řetězec smíšený nebo paralelní (Goughova-Stewartova plošina) viz obrázek 3 vlevo.

Běžnou praxí v robotice je popis řetězce pomocí sledu písmen vyjadřující typy kinematických dvojic, které obsahuje. Kupříkladu PPP, obsahuje tři posuvné kinematické dvojice - jde tedy o robot se 3 stupni volnosti kartézské struktury. Další příkladem je angulární struktura RRR se třemi rotačními KD.



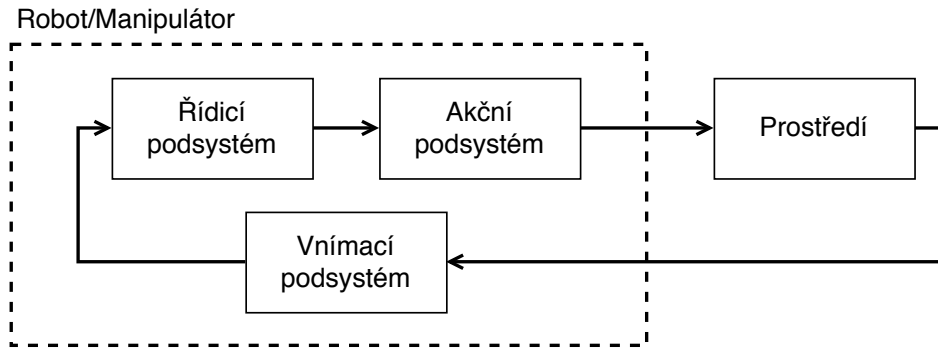
Obrázek 3: Srovnání paralelního [4] a seriového robotu

Zvláštní částí robotu je jeho **základna**, první článek kinematického řetězce, který je nehybně spojen se zemí. Na druhé straně řetězce pak **koncový efektor**, část uzpůsobená k interakci s okolím. Podle potřeby může být efektor buď uchopovací (často je pak nazýván chapadlo) nebo technologický, vybavený nástrojem jako je vrtačka, svářečka, nebo stříkácí pistole. Kinematickou soustavu robotu tedy zvláště tvoří tři složky - kinematika základny, ramen a koncového efektoru.

**Pracovní prostor** je množinou všech bodů, kterých může výstupní hlavička, čili koncový efektor, dosáhnout. Podle uspořádání robotu (RRR, RRP, PPP) se pak liší tvar obálky pracovního prostoru. Podle tvaru této obálky pak rozlišujeme kartézský, cylindrický a sférický typ pracovního prostoru a od nich se odvíjející souřadný systém. **Okolní prostor** robotu může obsahovat překážky kolize s nimi by mohla vést k poškození robotu nebo překážky, proto je potřeba předejít takové kolizi omezením pracovního prostoru robotu.

## 2.3 Struktura a členění

Strukturu manipulátor můžeme rozdělit do několika subsystémů.[5] Rozlišujeme subsystémy vnímání, řídicí a akční viz obrázek 4. Vidíme, že dohromady tvoří zpětnovazební systém s vazbou na okolní prostředí. **Vnímací subsystém** sestává převážně ze senzorů, kterými získává informace ze svého okolí, zároveň zde však spadá také veškeré vybavení umožňující vnitřní detekci různých, například chybových stavů. **řídicí subsystém** obsahuje části nezbytné pro řízení stroje a



Obrázek 4: Blokové schéma robotického manipulátoru

zpracování dat, spadá zde jak hardware, tak software. Patří sem zařízení pro uložení a vykonání programu, stejně jako samotný program, jeho algoritmy a data. **Akční subsystém** obsahuje prvky které můžeme pro potřeby dalšího popisu v praktické části dále rozdělit následujícím způsobem. [5]

- **Polohovací ústrojí** - K přemísťování obrobků (obvykle první tři členy struktury).
- **Orientační ústrojí** - K uvedení obrobku do specifické pozice (natočení, naklonění).
- **Koncový efektor** - K uchopování, či opracování obrobku (např. lakování).

**CNC** je zkratkou pro počítačové číslicové řízení - Computer Numeric Control. Původní zkratka NC, označovala zařízení, která narozdíl od mechanických a později elektromechanických metod řízení využívala číslicové obvody. S rozvojem počítačové techniky v druhé polovině minulého století začalo být výhodné využít ji také pro nahrazení specializovaných NC zařízení. Později s rozvojem CAD (Computer-Aided Design) pak bylo možné počítačem spravovat celý výrobní proces od návrhu až po jeho výrobu.

### 3 Prostředky řešení

Aby bylo možné efektivně navrhnout jak hardware, tak řízení manipulátoru, je třeba seznámit se s dostupnými komponentami a postupy, které mohou být využity. S ohledem na výše uvedené členění dílčích systémů manipulátoru je příhodné využít jej jako strukturu pro hlavní část teoretického úvodu. V následující části budou uvedeny různé komponenty, postupy a jejich srovnání vzhledem k požadavkům této práce. Takto budou vybrány vhodné prvky použité v praktické části.

#### 3.1 Řídicí subsystém

Dostupné materiály se prakticky dělí do dvou směrů, akademické robotiky a průmyslového CNC. Přestože jsou tyto směry v zásadě velice podobné, liší se zejména užitou terminologií a také přístupem k řešení úlohy. Metody řízení v robotice vyplývají spíše z matematického popisu a optimalizační úlohy někdy poskytují v praxi nerealizovatelná řešení. Na druhé straně CNC přístup vychází z průmyslové praxe a dbá na proveditelnost a výsledky. Jednodušší řízení sice nemusí být podle všech kritérií optimální, klade však nižší nároky na zařízení. [6]

Řízení můžeme zajistit pomocí různých systémů, mezi nejčastěji využívaná řešení patří programovatelné logické automaty, mikrokontroléry a průmyslové, případně osobní počítače.

##### 3.1.1 Programovatelné logické automaty

PLC jsou hojně využívány pro průmyslovou automatizaci a vyznačují se velmi vysokou spolehlivostí. Jedná se o uzavřená standardizovaná řešení, uzpůsobená pro průmyslové prostředí. Komfortní a efektivní vývoj je preferován za cenu vysokých nákladů, a proto se hodí pro automatizaci jednorázově vyráběných zařízení s požadavky na rychlý vývoj a spolehlivost.

Pro účely řízení pohybu s pomocí PLC byl organizací PLCopen vydán velmi rozšířený standard Motion Control. Tato norma sjednocuje většinu významných výrobců a umožňuje tak rychlejší a levnější vývoj méně závislý na konkrétní platformě. Jsou zde specifikovány zejména používané funkční bloky, procedury hledání počátku souřadnic tzv. „homing“, nároky na koordinovaný pohyb více os a další. Zbylé části programu jsou většinou vytvořeny pomocí jednoho z jazyků specifikovaných normou IEC61131-3.

I přes nesporné výhody rychlého vývoje a vysoké spolehlivosti, cena běžných PLC od výrobců jako Siemens či B&R mnohonásobně převyšuje rozpočet na stavbu stroje, a proto bude muset být dána přednost některé alternativě.

##### 3.1.2 Průmyslové počítače

Jsou zvláštní kategorií osobních počítačů, která se vyznačuje zvýšenou spolehlivostí a odolností k okolním vlivům jako je prašnost nebo vibrace. Vyznačují se zejména univerzálností. Jsou dobrým kompromisem mezi vestavěnými systémy a programovatelnými automaty.



U počítačů, ať již osobních, či průmyslových, je v kontextu řízení významným parametrem časování. Operační systém reálného času, často zkracovaný jako RTOS, je takový systém, který je programátorovi, potažmo uživateli, schopen zaručit, že je schopen poskytovat reakce v reálném čase, tedy průběžně. Tyto systémy definují časový rámec nutný k poskytnutí reakce zvaný deadline. Pokud je tento čas překročen, je výsledek jakékoli operace považován za nespolehlivý (tzv. soft-realtime), nebo přímo nebezpečný (tzv. hard-realtime). Požadavek na tuto vlastnost pak definuje výběr jak OS tak použitého software.

U méně kritických aplikací je velice často využíván OS Windows v kombinaci s programem MACH3. Dalším velice rozšířením řešením je využití LinuxCNC. Jedná se o distribuci OS Linux speciálně navrženou pro řízení CNC. K tomuto účelu zahrnuje také zvláštní RTOS nastavbu.

Výhodou těchto řešení je, že implicitně poskytují také HMI, nebo SCADA rozhraní, a stačí je před uvedením do chodu parametrizovat. Nevýhodou je však fakt, že jsou jen obtížně propojitelné s ostatními stroji a nelze je tak snadno rozšířit o další funkcionalitu k tomu potřebnou. Proto počítač v našem případě k samotnému řízení vybrán nebyl. Vzhledem k faktu, že stanice PC je však na pracovišti již přítomna, lze ji vhodně využít k realizaci uživatelského rozhraní viz kapitola 8.

### 3.1.3 Vestavěné systémy/mikrokontroléry

Mikrokontrolér, respektive vestavěný systém nebo prototypovací deska, které jej obsahují, jsou zařízení určená pro vysokou optimalizaci konkrétní aplikace. Vývojářův komfort je zde vyměněn za maximální upravitelnost a flexibilitu. Zároveň se vyznačuje velice nízkou cenou, tím pádem je vhodný pro automatizování hromadně vyráběných zařízení. Mnohé mikrokontroléry jsou distribuovány jako součást prototypovacích desek (Nucleo, BigClown, Arduino) usnadňujících jejich použití.

Právě vysoká flexibilita a nízké náklady vedly k tomu, že se použití těchto zařízení pro řízení pohybu 3D tiskáren značně rozšířilo. Naprostá většina 3D tiskáren je přitom jednoduchý RRR polohovací mechanismus s tryskou, neboli extrudérem na pozici koncového efektoru. Zejména pro potřeby tohoto odvětví se tedy během posledních let objevilo nepřeberné množství knihoven a programů, určených pro řízení kartézských a delta (alternativa s paralelní strukturou) zařízení. Mezi nejznámější patří projekty Marlin, Prusa nebo MakerBot.

Takový firmware je pochopitelně možné, po případné parametrizaci, použít pro jakékoli zařízení stejné struktury. A existuje řada programů a knihoven, které jsou buďto psány jako univerzální nebo se přímo specializují na zařízení jako jsou obráběcí frézky, plasmové či laserové routery. Mezi ně se řadí například firmware GRBL, Yaskawa nebo Reprap. Pro tento firmware většinou existuje řada dalších programů zprostředkovávajících HMI nebo SCADA.

Vzhledem k velice příznivým pořizovacím nákladům, vysoké flexibilitě, snadné rozšiřitelnosti a otevřenosti zdrojových kódů, bylo nakonec použito řízení za pomoci mikrokontroléru. Jeho výběr a implementace je blíže popsán v kapitole 5 v praktické části.

### 3.1.4 Programovací jazyky

Pro programování mikrokontrolérů se velice často používá jazyk symbolických adres. Obzvláště pro nejkritičtější části programu, kde je zapotřebí maximální efektivity, výpočetní rychlosti nebo práce s pamětí. Jazyk symbolických adres je nízkourovňový a je úzce spjat s hardwarem na kterém je spuštěn. Právě proto, že jeho úzké propojení s instrukční sadou procesoru vedlo ke vzniku mnoha odlišných druhů JSA, byly později vytvořeny jazyky univerzálnější s lépe srozumitelnou syntaxí jako například ANSI C.

**C** je jazykem používaným pro programování mnoha druhů zařízení včetně mikrokontrolérů po celá desetiletí. V 70. letech jej vytvořili D. Ritchie a B. Kernighan, jeho výhodou oproti JSA je, že se jedná o kompilovaný jazyk, díky čemuž je možné jej překládat pro práci na různých platformách a jeho syntaxe je bližší přirozenému jazyku, čili srozumitelnější. I v rámci kódu psaného v C, se obzvláště při programování mikrokontrolérů často setkáváme s částmi kódu v jazyce JSA, které jsou vloženy buďto v samostatných dokumentech, nebo takzvaně „inline“, čili přímo v kódu.

Jazyk C popisuje několik publikací a norem a různé odborné zdroje preferují různé verze jazyka, vycházejících z těchto pramenů. Mezi nejznámější patří ANSI C, podle normy z roku 1989, později ISO z roku 1990. Také původní verze jazyka často přezdívaná „K&R“ podle tvůrců tohoto jazyka a zároveň autorů knihy „The C Programming Language“, ve které byl poprvé popsán. [7] Posledním příkladem může být standard MISRA, vyvinutý pro bezpečné aplikace v mikrokontrolérech využívaných v automobilovém průmyslu. MISRA-C vychází z normy ISO 9899:1990[2] a předkládá seznam povinných pravidel a doporučení pro programování v jazyce C. [8]

**C++** bylo vyvinuto Bjarne Stroustrupem nedlouho po vydání ISO-C. Rozšiřuje C o prvky objektově orientovaného programování. Jazyky jsou tedy velice podobné a prakticky vzato kód v jazyce C je vždy možné bez problémů použít při programování v C++, naopak to možné není kvůli přidaným funkcím C++, které by C kompilátor nebyl schopen zpracovat.

**Python** v poslední době získává na popularitě v mnoha oborech a mimo jiné při programování mikrokontrolérů. Narozdíl od JSA a C/C++ jde o jazyk interpretovaný, takže na cílové platformě je zdrojový kód prováděn pomocí programu zvaného „interpret“. To umožňuje ještě snadnější úpravy kódu a ladění než jazyk C, vykoupené o něco nižší rychlostí. Podobně jako u kompilovaných jazyků, pokud existuje na pro danou platformu interpret, můžeme kód na této platformě spustit. Python je vysokoúrovňovým jazykem, zcela zásadní je objektově orientované paradigma. Obsahuje a podporuje podobně jako C/C++ řadu modulů a knihoven pro urychlení práce jako „pickle, urllib.request a tkinter“ pro (v uvedeném pořadí) ukládání datových struktur, pro čtení a práci s webovými stránkami a pro práci s grafikou a uživatelskými rozhraními. [9]

Pro použití v aplikacích s mikrokontroléry byl navíc vyvinut **Micropython**, implementace Pythonu3 jako kompilátoru a běhového prostředí optimalizovaného pro použití přímo na mikro-

kontrolérech. Umožňuje programování pomocí tzv. „REPL“, interaktivního příkazového řádku podporujícího historii příkazů, automatické doplňování kódu a také vybrané knihovny Pythonu.

V této práci bylo použito celkem 5 programovacích jazyků, dva výše zmíněné, C/C++ k programování použitých MCU a další dva při tvorbě uživatelského rozhraní, které jsou detailněji rozebrány v kapitole 8.

Pro programování řízení bylo použito C (13 % kódu v uložisti) a C++ (46 %), zbylá procenta náleží programům uživatelského prostředí (údaje byly vypočteny repozitářem na Github). Větší část je tedy napsána v jazyce C++. Jazyk C byl využit pro napsání nízkourovňových funkcí, umožňujících lepší práci s pamětí při ukládání zejména textových řetězců. Tato funkcionality byla implementována, aby bylo možné ukládat přímo v programu velké množství textových instrukcí G-kódu, kterými bude později ovládán pohyb robotu.

G-kód, někdy také RS274 je standardizovaný programovací jazyk používaný pro ovládání CNC strojů. Skládá se z jednoduchých instrukcí získaných jako kombinace písmen a hodnot. Název jazyka je odvozen od pojmu „g-kód“, který označuje sadu přípravných kódů začínajících písmenem G. Tato sada obsahuje příkazy, které jsou při programování používány nejčastěji, jako například „G1“ pro lineárně interpolovaný pohyb. Ukázka g-kódu je uvedena níže.

---

```
G0 Z5.0000
X161.9241 Y170.1218
Z3.0000
M3
G4 P0.5
G1 X161.3685Y171.2847F1200.0
X147.8406Y184.8126
X103.2000Y140.1720
Y123.5092
X147.8406Y78.8686
X161.3685Y92.3966
M5
G0 Z5.0000
```

---

Výpis 1: Ukázka G-kódu, obrys vnitřní rohové vzpěry

## 3.2 Vnímací subsystém

Tato podkapitola se zabývá teoretickým rozбором možných senzorů, které by bylo možné implementovat a srovnáním jejich výhod a nevýhod. Sensory můžeme využít zejména pro získávání informací z pracovního i okolního prostoru. Můžeme pomocí nich vyhodnocovat, například zda se v pracovním prostoru nenachází nežádoucí objekty a tělesa, zda je cyklus prováděn správně (obrobek byl skutečně uchopen, má správnou barvu atp.) a také kontrolovat vnitřní stav stroje (zda motory neztratily pozici, nedochází k přehřívání CPU).

### 3.2.1 Snímače limitní polohy

Obyčejně bývají zastoupeny 2 typy senzorů lišících se fyzikálním principem snímání. Základním a velice často používaným typem senzoru jsou kontaktní koncové spínače. Tyto spínače bývají vybaveny tzv. mžikovým spínáním, kdy i při velmi pozvolném působení na spínací mechanismus dojde ke krokovému přepnutí z jednoho stavu do druhého. A to pokud možno pokaždé ve stejném bodě. Stejným způsobem dochází také k rozpínání. Doteková část těchto snímačů má řadu variant, většinou se jedná o zásuvný, pružný nebo otočný mechanismus. Typické pracovní napětí těchto snímačů bývá 5-24 VDC, proud se pohybuje řádově v desítkách miliampér, a to z důvodů minimalizace jiskření na kontaktech. To by vedlo k jejich opalování a snižování životnosti kontaktů. Mezi nejznámější výrobce těchto spínačů se řadí například firmy Omron, Honeywell nebo Schneider.

Druhou často využívanou možností, jsou snímače měřící magnetické vlastnosti, jmenovitě indukční senzory. Jde o aktivní snímače měřící tlumení vestavěného oscilátoru v důsledku přítomnosti magneticky vodivého materiálu v jejich blízkém okolí. Typické pracovní napětí se pohybuje mezi 10-30 VDC a proud v sepnutém stavu řádově v desítkách miliampér. Nejčastější je válcové šroubové provedení se dvěma uchycovacími maticemi. Jejich hlavní nevýhodou je nezbytnost napájení, výhodou je fakt, že častým používáním nedochází k jejich fyzickému opotřebení. [10] Tyto senzory mohou být využity jako koncové spínače; můžeme s nimi však také zjišťovat jak správnost přichycení dílu k magnetické hlavě manipulátoru, tak přítomnost hotových dílů na odkládací paletě.

### 3.2.2 Přítomnostní snímače

Jak bylo zmíněno výše, mohou fungovat na principech magnetismu, za předpokladu, že obrobek je kovový. V případě této práce samozřejmě předpokládáme, že je. Pokud bychom chtěli hledat i jiné předměty, například kontrolovat, zda je přítomna odkládací dřevěná paleta, bylo by vhodnější vybavit koncový efektor snímačem, umožňujícím nalezení nekovového předmětu. Takovým požadavkům vyhovuje například optický snímač vzdálenosti. Tyto snímače často umožňují kalibrovat vzdálenost, na kterou jsou zaostřeny. Pokud se překážka objeví v této vzdálenosti, výstup snímače přejde do patřičného logického stavu.

### 3.2.3 Bezpečnostní snímače

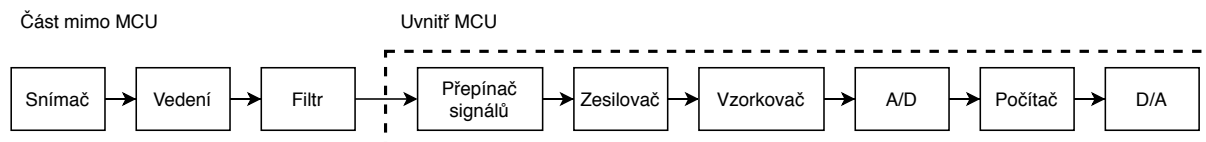
jsou poslední skupinou, která zde bude uvedena. Roboty a manipulátory, které nejsou přímo určeny pro spolupráci s člověkem (kooperativní roboty), většinou neimplementují takové množství senzorů, aby se mohly bezpečně vyhnout pohyblivému objektu ve své pracovní ploše. Pohyb v okolí takového robotu může být značně nebezpečný. Toto nebezpečí snadno vyloučíme použitím bezpečnostních snímačů omezujících vstup na pracoviště. Pokud by došlo ke vstupu operátora do pracovního prostoru, uvede se přístroj vhodným způsobem do klidového režimu. Do této kategorie snímačů patří zejména optické brány a nášlapné panely. Optické brány se skládají ze dvou částí, mezi nimiž vytvoří řadu paprsků, a jejich přerušení je detekováno. Nášlapné panely obvykle fungují jako řada mechanických rozpínačů bez účasti dalších vyhodnocovacích obvodů.

### 3.2.4 Elektromagnetické rušení

EMC, čili elektromagnetická kompatibilita je definována jako schopnost zařízení nepodléhat rušivým vlivům okolí, stejně tak jako nevytvářet svým provozem elektromagnetické rušení ve svém okolí. Elektromagnetické rušení, v průmyslovém prostředí prakticky není možné zcela potlačit, protože má zásadní negativní dopad na přesnost všech prováděných měření, je třeba je omezit na nezbytné minimum. Existuje celá řada způsobů jak aktivně, tak i pasivně potlačovat vlivy elektromagnetického rušení. Tyto metody se liší v závislosti na tom, které části měřicího řetězce se týkají.

Samotný snímač je méně náchylný na chyby, včetně chyb způsobených elektromagnetickým rušivým signálem, využijeme-li například metodu diferenčního snímače. Princip této metody předpokládá, že při měření dvěma shodnými snímači, připojenými na diferenční zesilovač, přichází rušivý signál na oba vstupy se stejným znaménkem, kdežto užitečný signál vstupuje do zesilovače se znaménkem opačným. V takovém případě je rušivý signál jednoduše odečten.

Částí měřicího řetězce (lze vidět na obrázku 5), kde dochází k ovlivnění signálu rušením nejčastěji, je signálové vedení. To je zatíženo zejména galvanickou, v závislosti na délce kapacitní a v závislosti na protékaném proudu, indukční vazbou. Pro potlačení indukční vazby se využívá kroucení párů vodičů, čímž je snížena vzájemná indukčnost. Kapacitní vazbu je možné potlačit vhodným uzemněním zdroje rušení nebo stíněním uzemněným na jednom z konců stínění. Galvanickou vazbu pak potlačujeme oddělovacím transformátorem nebo optočlenem. V neposlední řadě je potřeba zhodnotit, zda má přenášený signál proudový nebo napěťový charakter. Napěťový signál je náchylnější k rušení a je vhodné používat vyšší hladiny napětí, aby nedocházelo k snadnému ovlivnění. Běžně používané hladiny napětí jsou 12 a 24 VDC. Mnohem méně náchylný je však systém využívající proudový signál. Běžnou hodnotou unifikovaného proudového signálu je 4-20 mA, nevýhodou je náročnější vyhodnocování na vstupu MCU.



Obrázek 5: Blokové schéma měřícího řetězce

### 3.3 Akční subsystém

Aktuátor neboli akční člen je část mechatronické soustavy, kterým řídicí systém působí na okolní prostředí. Pro řešení úlohy je podstatné zejména obeznámit se různými typy pohonů, které jsou v robotice využívány.

#### 3.3.1 Pohony

V průmyslu je využívána široká škála rozličných typů pohonů, mechanické, elektrické, pneumatické a hydraulické. Zejména v robotice jsou však nejčastěji využívány elektrické pohony, které poskytují řadu výhodných vlastností. Elektrických pohonů však existuje celá řada nejčastěji se však setkáváme se střídavými servopohony.

#### Střídavé motory

Nejběžnější typ, asynchronní motor s kotvou nakrátko, jsou velice jednoduché a spolehlivé zařízení. Pokud není potřeba velkých výkonů, používají se jednofázové motory. Pro usnadnění rozběhu jsou často vybaveny pomocnou fází s kondenzátorem (ten zabezpečuje potřebný fázový posun proudu) a odstředivým rozpínačem, který pomocnou fází po rozběhu odpojí. Tyto motory často bývají vybavené elektromagnetickými brzdami a reverzačními spojkami. Rychlost otáčení je nejčastěji regulována změnou frekvence napájení, k čemuž jsou využívány polovodičové měniče. Vybavením motoru zpětnovazebním snímáním polohy můžeme asynchronní motor využít jako servo. Vhodnou regulací můžeme precizně ovládat úhel natočení nebo rychlost otáčení hřídele. Výhodou těchto motorů je příznivá cena, bezúdržbovost a vysoká momentová zatížitelnost.

Obdobným způsobem se v poslední době často využívají synchronní motory, které sice vyžadují cizí buzení a jsou složitější konstrukce, ale vyznačují se vysokou spolehlivostí a jejich řízení je ve srovnání s indukčními (asynchronními) motory jednodušší. Obecnou nevýhodou servopohonů je jejich vysoká cena.

Nevýhodou střídavých pohonů je zejména obtížnější regulace a to jak rychlosti, tak i polohy. Přestože jsou polovodičové měniče dnes mnohem dostupnější než dříve, jsou stejně jako serva nadále velice nákladné v porovnání s níže uvedenými typy pohonů.

## Krokové motory

jsou motory, které se synchronně otáčejí v závislosti na spínání dvojic vinutí statoru. Toto otáčení není spojité, motor přeskakuje mezi jednotlivými pozicemi - kroky. Spínáním kombinací přilehlých vinutí můžeme zmenšit rozlišení úhlu otočení - mikrokrokování. Krokové motory se vyznačují snadným přesným řízením v otevřené smyčce. To je typicky realizováno přiváděním napěťových impulsů na ovladač, často nazývaný anglickým termínem driver, který je zesiluje a převádí postupně na jednotlivé vinutí statoru. Tyto motory se vyznačují poměrně nízkými kroutivými momenty.

## Hybridní servo

je kompromisem výše uvedených pohonů. Jde v praxi o obchodní název označující řízení krokového motoru v uzavřené smyčce. [11] Tímto jednoduchým způsobem je možné získat přesné řízení s minimálními pořizovacími náklady ve srovnání s jinými typy servopohonů, jelikož nároky na řídicí techniku jsou také nižší. Tyto pohony bývají realizovány jako dvojice motor-ovladač, kdy motor je již od výroby vybaven rotačním encodérem připevněným na hřídel. Ovladač jednoduše porovnává počet vyslaných a přijatých impulsů a pokračuje v otáčení dokud se tyto hodnoty nerovnájí. Pokud se mu nepodaří v přijatelném čase regulační odchylku eliminovat, zastaví pohyb a na příslušném výstupu signalizuje chybový stav. Z výše uvedených důvodů tento typ pohonu získává na popularitě zejména v oblasti výroby CNC.

### 3.4 Koncový efektor

Často nazývaný také výstupní efektor, výstupní hlavice nebo pracovní hlavice, tato speciální označení vyplývají většinou z typu uchopovacího mechanismu. Typy výstupních hlavic jsou v zásadě tři. Úchopná, technologická a měřicí. Manipulátor, kterým se tato práce zabývá má být uzpůsoben zejména k přemísťování a mezioperační manipulaci s obrobky. K tomuto účelu se užívají hlavice úchopné. Aktivní úchopné prvky jsou typicky buďto mechanického, elektromagnetického nebo pneumatického typu.

Magnetické úchopné hlavice jsou výhodné zejména z těchto důvodů. S její pomocí lze snadno uchopit díly se značně rozdílnými rozměry a členitým povrchem (profilovaný plech), které se obtížně uchopují pneumaticky a zároveň je schopný zvednout rovný plech z podložky, který by se obtížně uchopoval mechanickým efektozem.

Mají však také nevýhody. Zaprvé se k nim přichytává také struska a kovové piliny, které po nashromáždění mohou zhoršit kontakt s obrobkem a tím snižuje pevnost přichycení. Za druhé uchopený předmět se zmagnetizuje, kvůli čemuž následně nemusí dojít k jeho odpadnutí při upuštění. To je možné vyřešit krátkým obrácením směru proudu v magnetech, bezprostředně po přepnutí do puštěného stavu, čímž dojde k vyrušení remanentního magnetického pole a obrobek snadno odpadne. [12]

## 4 Integrace do výrobního procesu

Jak již bylo zmíněno v úvodu, cílem této práce je automatizovat proces výroby dílů ohradové palety, tento proces byl již před začátkem práce mechanizován. Fyzická lidská práce již byla nahrazena stroji, naším úkolem je však proces automatizovat tak, aby zásah člověka pro samotnou výrobu dílů nebyl vůbec nutný.

V současné době jeden pracovník nemůže jednoduše vyrábět více než jeden typ dílu. Tento pracovník navíc značné množství času tráví přenášením dílů mezi jednotlivými stroji. Kdyby se operátor nemusel zabývat přenášením dílů, mohl by se spíše věnovat práci, pro kterou je specificky potřeba lidské asistence, jako výměny spotřebních dílů nebo doplňování výrobních materiálů. Přidáním manipulátoru by mohl být celý proces automatizován tak, aby se ovládání celého systému omezilo na zadání požadovaného počtu vyrobených dílů.

### 4.1 Popis výrobního procesu

Před začátkem práce jsou, do k tomu určených odvíjecích podstavců, založeny cívky s plechem o požadované tloušťce 1,5 a 4 mm. Z těchto podstavců je plech založen do rovnacích strojů, které zároveň slouží k podávání narovnaných plechů do dalších částí výroby.

V případě, že jde o 4mm plech, je odvíjen pomocí válečkového dopravníku přímo do plasmového routeru, kde je vyřezán tvar dílu nohy. Při řezání plazmovým hořákem v neideálních podmínkách, jako je zvýšená vzdušná vlhkost, dochází k tvorbě snadno odlomitelné strusky zejména na spodním okraji dílu. Každý díl je tedy ručně odebrán, zbaven strusky a umístěn do sloupců po pěti kusech na skladovací palety. Později jsou tyto palety přivezeny k lisu, kde jsou díly naohýbány do pravého úhlu a znovu stohovány tak, aby do nich mohly být navařeny vzpěry.

V případě, že jde o plech o tloušťce 1,5 mm, je rovnacím zařízením podáván přímo do hydraulického lisu, kde je vhodným způsobem naohýbán požadovaný profil dna nebo bočnice. Tento profil, na konci rovného plechu, je pak po kolejnicích posunut skrz lisovací rám, až do prostoru plasmového routeru. Kde je přichycen na požadované pozici a odřezán. Poté se zbytek plechu vrací zpět před rám lisu, aby mohl být naohýbán další díl. Hotové díly zůstávají v prostoru plasmy, odkud jsou ručně odneseny. Kratší strany palety jsou vyráběny se stejným profilem jako delší strany, navíc jsou plazmovým řezákem později zkráceny na patřičný rozměr. Vyrobené díly se stohují na další odkládací palety.

Všechny výše uvedené díly jsou pak postupně ručně svařeny. Výsledné bedny, jsou pak namáčeny do barvy dalším manipulačním mechanismem, ten však není součástí této práce a z bezpečnostních důvodů je umístěn na jiném, lépe větraném pracovišti.



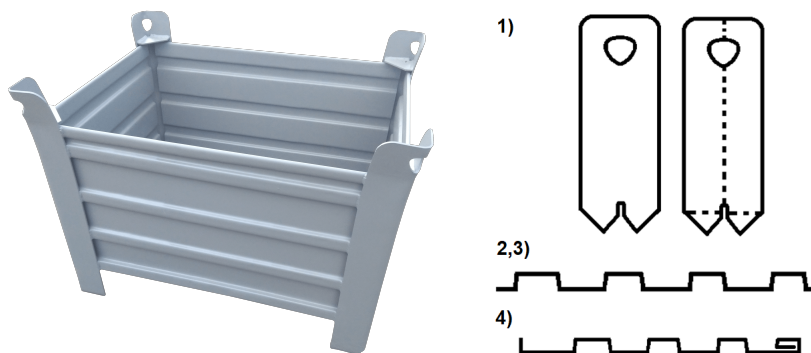
## 4.2 Ohradová paleta

je kovová přepravní, stohovatelná bedna s vyztuženým kovovým dnem, nejčastěji používaná ve strojírenském průmyslu. Jedná se o vratný obal určený k vícenásobnému použití a je tedy mimo jiné šetrnější k životnímu prostředí. Je navržena a uzpůsobena pro práci s paletovým i vysokozdvizným vozíkem, stejně jako k zavěšení pod různé typy jeřábů. Velikou výhodou je také její stohovatelnost. Typicky je využívána pro přepravu nebo vnitropodnikové uskladnění kovových obrobků. Tento obal je dostupný v řadě různých variant, jelikož většina výrobců se od originální dokumentace, která je mimo základní rozměry jen velmi obecná, drobnými úpravami odlišuje. Často používané jsou jména Sfinx a MARS, obě však v praxi označují ohradovou paletu s vyztužením.

Přesto je možné rozdělovat dva základní typy, paletu velkou a malou. Jejich parametry jsou blíže rozvedeny v tabulce 1. Z těchto parametrů vyplývá, že rozměry jsou úmyslně takové, aby odpovídaly celé nebo poloviční velikosti palety EUR dle normy ČSN 26 9110 (Evropská dřevěná čtyřcestná prostá paleta).

Tabulka 1: Parametry různých typů palet MARS

Typ palety	Hloubka (cm)	Šířka (cm)	Výška (cm)	Hmotnost (kg)	Nosnost (t)
MARS Velká	80	120	60	57	1,5
MARS Malá	60	80	60	40	0,75



Obrázek 6: Malá ohradová paleta a její části

V souladu s výše uvedeným znázorněním na obrázku 6, se paleta skládá z těchto dílů:

1. Rohová noha s dvojítm ohybem, znesnadňujícím její uchopení manipulátorem.
2. Dno s vysokým profilem, jehož velká váha ovlivňuje dynamiku manipulace.
3. Bok dlouhý s přehybem, jehož časově náročný cyklus ovlivňuje plynulost výroby.
4. Bok krátký s poloviční délkou, na kterou je potřeba jej krátit až po ohnutí.

### 4.3 Výchozí vybavení strojů

Před začátkem práce byla mechatronická úloha složena z několika strojů, které byly řízeny vývojovými deskami typu Arduino. Jednotlivé stroje spolu byly schopny jednoduše komunikovat pomocí sběrnice I<sup>2</sup>C a vyměňovat si tak například údaje z některých senzorů. Veškeré elektronické komponenty byly umístěny v jednom rozvaděči vybaveném potřebnými zdroji, svorkovnicemi a ovladači. Pohony byly zajištěny krokovými nebo třífázovými motory s vhodnými měniči, umístěnými mimo skříň rozvaděče. Bezpečnost zajištěna přidáním vhodných prvků jako tlačítek nouzového zastavení a bezpečnostních snímačů, které lze vidět např. na obrázku 14 a fyzickými zábranami, omezujícími možnosti styku člověka s pohyblivými částmi stroje.

### 4.4 Návržení vhodných úprav

Řízení strojů bude přesunuto na mikrokontroléry, ATMega2560 a ATMega328p, díky kterým půjde snadno přenést kód z původních strojů. Tyto MCU budou osazeny na deskách lépe uzpůsobených k provozu v průmyslu, podporujících sběrnici CAN. MCU řídící manipulátor bude níže detailněji popsán. Veškerá komunikace strojů bude přesunuta na sběrnici CAN a bude zaveden jednoduchý universální komunikační protokol. Jak manipulátor, tak ostatní stroje budou osazeny více snímači lépe přizpůsobenými rušení.

Manipulátor bude mít svou elektroniku umístěnou ve vlastním rozvaděči, disponujícím samostatným zdrojem napětí. Bude řízen výše uvedenými MCU komunikujícími mimo rozvaděč po sběrnici CAN a mezi sebou po sériové lince. Toto MCU bude možné přeprogramovat seriovým programátorem. Bude se tedy jednat o samostatný stroj, nezávislý na provozu ostatních strojů, ale zároveň s nimi propojený a schopný komunikovat a spolupracovat.

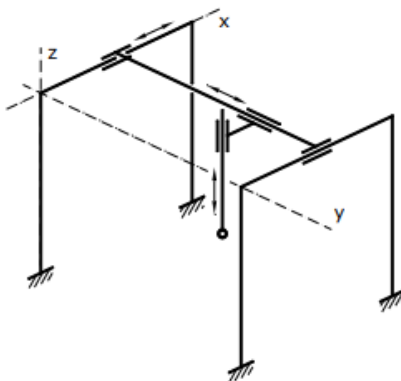
## 5 Realizace manipulátoru

Tato kapitola rozebírá zejména jakým způsobem byl manipulátor sestaven, jaké byly použity komponenty, a jakým byly použity způsobem. Samotnými fázemi montáže a programování, kterými manipulátor prošel před svým spuštěním a jaké změny byly aplikovány po proběhnutí prvního testování, jsou rozebrány v kapitole 6.

### 5.1 Popis mechaniky

Mechanická část již sice byla sestavena před začátkem této práce, je však nezbytné uvést zde jakým způsobem byla navržena, jelikož její návrh do značné míry ovlivňuje veškeré další použité komponenty a jejich výběr. Polohovací a orientační ústrojí byly řešeny takto.

Pracovní prostor, ve kterém má manipulátor pracovat je rozměrově poměrně rozsáhlý ( $8 \times 3$  m), jelikož koncový efektor musí být v dosahu odkládacích stanovišť všech strojů zapojených do výrobního procesu. Začleněním vhodného polohovacího mechanismu, například kolejnice, může sice být rozšířen dosah kteréhokoli typu robotu, například typu RRR, takový přístup by však z řízení zbytečně komplikoval řešení. Ideální se zdál být kartézský systém struktury PPP, který je jednoduše realizovatelný stejně jako říditelný.



Obrázek 7: Kinematická struktura PPP manipulátoru

Osa X, umožňující pohyb s největším rozsahem je řešena kolejnicemi, na jejichž stranách je nepružné ozubení. Po dvou těchto kolejnicích je vedeno příčné rameno osy Y, na němž se nachází posuvný vozík umožňující pohyb v této ose. Obě tyto osy jsou ovládány krokovými motory, které bude v případě potřeby vyšší spolehlivosti možné snadno nahradit hybridními servo motory. Každý z těchto pohonů je připojen k ozubenému kolu, které s patřičným převodem zprostředkovává v kombinaci s ozubením kolejnice X, případně řemenem osy Y, translační pohyb.

Osa Z je řešena tuhým teleskopickým ramenem, aby byly zajištěny potřebné dynamické vlastnosti, zejména omezeno kmitání. Zdvih této osy je zajištěn dalším elektrickým motorem (stejněho typu, rozdílného výkonu), a to skrze navíjecí buben, který spodní část teleskopu vytahuje ve směru osy Z pomocí soustavy kladek.

Jelikož výrobní proces vyžaduje přesnější manipulaci a to konkrétně v 5 stupních volnosti, je koncový efektor manipulátoru vybaven orientačním ústrojím pro zajištění pohybu v těchto dodatečných dvou osách. První stupeň orientačního ústrojí je kinematickou dvojicí rotačního typu (R), a zajišťuje otáčení v horizontální ose. Druhý stupeň je mechanismus zabudovaný v ose Z, který umožňuje naklápění koncového efektoru v rozsahu přibližně  $180^\circ$  ve vertikální ose. Při konstrukci manipulátoru bylo přijato označení této osy písmenem N (náklon), rotační mechanismus stejným způsobem získal označení R (rotace).

Samotné uchycení dílu je zprostředkováno pomocí 4 elektromagnetů, ty jsou k hlavě manipulátoru pružně uchyceny tak, že se mohou náklonem mírně přizpůsobit tvaru dílu a zajistit tak bezpečné a pevné spojení.



Obrázek 8: Koncový efektor se 4 elektromagnety [12]

## 5.2 Pohony

Moderní řídicí systémy regulují nejen polohu, ale i rychlost, zrychlení a derivaci zrychlení, tzv. „jerk“. Zejména poslední dvě z výše uvedených mají neblahý vliv na stabilitu mechanické konstrukce, nejsou-li řízeny správně. Rychlé změny ve směru a velikosti zrychlení vybuzují v mechanické soustavě kmitání v okolí rezonančních kmitočtů této soustavy. Kmitání manipulátoru je pochopitelně nežádoucí, zvláště u manipulátorů velkých rozměrů, kde může být rozkmit na konci ramene v rozsahu až několika centimetrů. Toto můžeme řešit třemi způsoby. Moderní roboty a manipulátory využívají nejčastěji různých typů serv. Servo zajistí, že nedojde ke ztrátě polohy a vhodným způsobem reguluje její derivace pro dosažení co nejlepších výsledků. Druhou možností je zvýšit tuhost konstrukce a tím zamezit ztrátě přesnosti v průběhu pracovního cyklu.

Oba výše uvedené způsoby jsou však značně nákladné. Třetím způsobem je vhodně omezit maximální hodnoty zrychlení a rychlosti. V kombinaci s dostatečně robustní konstrukcí je pak

možné s dostatečnou přesností a spolehlivostí řídit pohyb v otevřené smyčce pomocí krokových motorů. Poněkud nadneseně, je možné považovat za výhodu také fakt, že vhodně dimenzovaný krokový motor v případě kolize s jiným objektem jednoduše ztratí pozici, nebude však působit dostatečnou silou, aby objekt poškodil. Zejména ve fázi prototypování má však tato skutečnost v praxi velmi kladný vliv na výslednou bezpečnost.

Jak již bylo zmíněno v teoretické části, zejména v oblasti CNC řízení je moderním trendem pro aplikace, kde není důležitá vysoká výkonová přetížitelnost (jako frézování), využívat krokových motorů v uzavřené smyčce. Jejich ovladače se chovají navenek stejně jako obyčejné ovladače krokových motorů a v samotném pohonu je pouze přidána část s enkodérem. Prototyp manipulátoru byl postaven s použitím krokových motorů a obyčejných driverů. Po uvedení do provozu však bude možné v případě potřeby velmi jednoduše zvýšit přesnost a spolehlivost zařízení namontováním lepších měničů.

### 5.3 Senzory

Nevýhodou krokových motorů je neznalost absolutní polohy, takže po zapnutí stroje není žádný způsob jak zjistit, kde se zrovna nachází a je nutné nalézt referenční polohu. Hledání výchozí pozice je zajištěno snímačem limitní polohy na konci pojezdu. Pokud je koncový snímač mechanický, největší nároky jsou kladeny na jeho mechanickou odolnost. Jelikož chybí údaj o vzdálenosti, kterou je třeba ke snímači urazit, přibližuje se manipulátor při hledání koncových snímačů pomaleji, aby k nárazu nedošlo na plné přejezdové rychlosti a možnému poškození snímače.

Dále byl použit přítomnostní senzor, kterým je detekováno, zda se elektromagnetům podařilo uchytit požadovaný díl. Takový senzor může být založen na převodu různých fyzikálních veličin. Mohli bychom využít například optický, kapacitní nebo indukční senzor, každý disponující různými výhodnými vlastnostmi, z nichž vzhledem k tomu, že všechny diskutované díly jsou kovové, se jako nejvhodnější jeví indukční snímače. Na druhou stranu uchopovací mechanismus obsahuje silné elektromagnety, mohlo by tedy dojít ke vzájemnému ovlivnění. Dostupné optické přítomnostní senzory však umožňují nastavit pouze jednu snímací vzdálenost. Jako nejvýhodnější tedy přece jen jeví indukční snímač.

Tlačítko total-stop, často také tlačítko nouzového zastavení, je nejzákladnějším a neodmyslitelným prvkem každého průmyslového zařízení. Bezpečnost včetně použití těchto tlačítek řeší normy ČSN EN 60204 (Bezpečnost strojních zařízení) a ČSN EN ISO 13849 (Bezpečnostní části ovládacích systémů). [13] Toto tlačítko je možné zapojit několika různými způsoby, které se odvíjejí od bezpečnostní kategorie, do které zařízení spadá. Není je vhodné seriově spojovat a běžně jsou zapojovány s redundantními spínacími prvky i vodiči pro maximální omezení možnosti poruchy. Vhodné je zapojit jej tak, aby bylo pro obsluhu snadno dostupné, bylo tedy připojeno na skříň rozvaděče tak, aby po jeho sepnutí došlo k odpojení všech pohyblivých částí od zdroje napájení.

## 5.4 Rušení

Jeden ze strojů, který má být manipulátorem obsluhován je stůl pro plazmové řezání. Plazmová řezačka, respektive její trafo je však problematickým zařízením z hlediska EMI. Plazmové řezačky, používají k opracování materiálu elektrický oblouk. Zažehnutí tohoto oblouku probíhá bezkontaktně, namísto doteku a oddálení elektrod totiž řezačky využívají vysokofrekvenčních zážehů. Velká napětí a vysoké frekvence potřebné pro zážeh oblouku jsou však významným zdrojem rušení všech okolních zařízení. Ve snaze odstranit neblahé vlivy tohoto rušení byla postupně iterovány tyto opatření:

**Pull-up rezistory** – pro čtení I/O vstupu mikrokontrolérem jsou samozřejmě tyto rezistory nezbytné, jelikož vstup je v režimu vysoké impedance. Přidáním paralelního odporu se však výsledný odpor zmenší a vstup je „přitahován silněji“. Pochopitelně toto opatření mělo zanedbatelný účinek.

**RC-filtry** – Přidáním vhodného kondenzátoru vytvoříme jednoduchý dolnoproustný filtr, který rušivé vysoké frekvence utlumí. Složitější filtry s přesnějšími frekvenčními charakteristikami pak byly přidány na napájení, pro případ, že rušení se přenáší také napájecí sítí. Toto opatření již přineslo mírné zlepšení.

**Schmittovy klopné obvody** - jsou jednoduché polovodičové obvody typicky ve formě DIP integrovaných čipů. Hlavní vlastností těchto obvodů je hystereze, která zajistí v podstatě další úroveň filtrace. Zajišťuje, že rušivý signál musí překročit hranici citlivosti obvodu, aby došlo k překlopení výstupu, což v kombinaci s vhodným RC filtrem poskytuje poměrně dobrou úroveň ochrany. Překvapivě v praxi toto nepřineslo očekávané zlepšení.

**Úroveň signálu** - V průmyslovém prostředí samozřejmě nebylo možné vést signál na úrovni 0-5 V, proto byla použita standardnější a robustnější hladina 0-24 V. Takové napětí však nelze na MCU přímo číst, a proto musí být převáděna na nižší úroveň a to externím obvodem, v nejjednodušším případě děličem napětí.

**Galvanické oddělení** – Velký podíl na přenosu rušení mívá galvanická vazba, je tedy vhodné všechny důležité části galvanicky oddělit. Většina průmyslových zdrojů je již od výroby galvanicky oddělena. Slabým článkem je zde spojení s PC, proto je vhodné galvanicky oddělit zejména veškeré zařízení spojené pomocí USB, jako jsou programátory. Toto opatření se ukázalo být účinným.

**Dolnoproustný filtr** – Vzhledem k vysokofrekvenční povaze rušení bylo logickým řešením přidání dolnoproustného filtru s frekvencí zlomu danou vztahem 1, dosazené hodnoty se různí v závislosti na frekvenci výstupu konkrétního snímače.

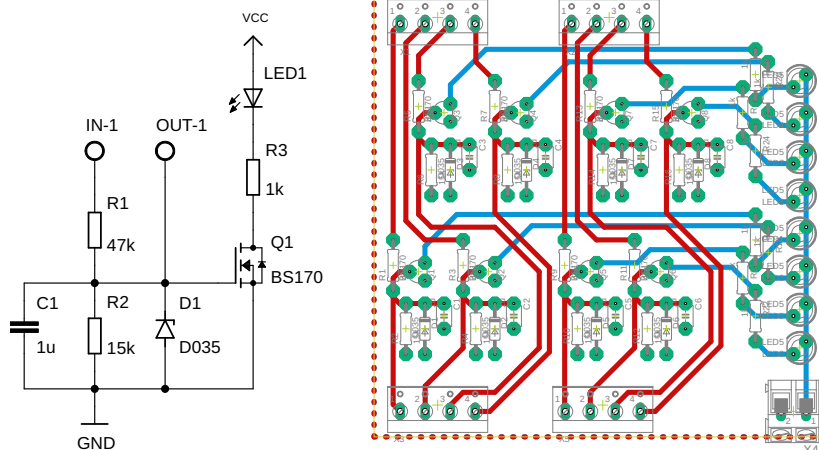
$$f_c = \frac{1}{2\pi RC} \quad (1)$$

## Vytvoření obvodu pro úpravu signálu

Po zavedení výše uvedených opatření začalo být možné číst digitální vstupy spolehlivě. Tyto prvky pochopitelně musely být centralizovány, aby bylo možné je namontovat do rozvaděče. V prototypovací fázi práce proto byla navržena a osazena deska plošných spojů s vhodným obvodem. Po otestování v provozu, byla ve spolupráci s Ing. Golembiovským navržena druhá vylepšená verze, využívající SMD technologii, šetříc tak zastavěný prostor v rozvaděči o více než polovinu.

Obvod uvedený na obrázku 9 vlevo, sestává z děliče napětí, stabilizační zenerovy diody a kondenzátoru. Základní vztahy pro výběr těchto komponent jsem čerpal z vhodné literatury. [14] V praxi se ukázala být velice přínosná možnost indikovat sepnutý stav, a proto byla přidána indikační dioda a spínaná unipolárním tranzistorem [15]. Obrázek 9 zobrazuje první verzi DPS. Výsledný návrh DPS měl veškeré plochy vylité mědí, jelikož deska byla vytvořena pouze jako prototyp bez dokumentace, je tento obrázek pouze ilustrativní a pro přehlednost bylo vyplnění polygonu vynecháno. První verze této desky je s  $80 \times 80$  mm poměrně rozměrná, prioritou však byla robustnost a snadná výrobitelnost.

Druhá verze vznikla ve spolupráci s Ing. Golembiovským, využívá SMD součástky, je rovněž dvouvrstvá a přinesla značné zmenšení rozměrů z původních  $80 \times 80$  mm na  $60 \times 42$  mm. Tato verze bude zaslána profesionálnímu výrobcí a použita v reálném provozu po dokončení testování. V příloze A jsou přiloženy výkresy obou vrstev pro tuto verzi.



Obrázek 9: DPS s obvodem pro čtení I/O vstupů

## 5.5 Rozvaděč

je z hlediska napájení vybaven zdrojem 24 V a zásuvkou 230 V pro připojení zařízení užitečných například při opravách diskutovaného rozvaděče. Pro zařízení s napájecím napětím menším než 24 V stačí použít step-down měnič.

Dále je potřeba použít vhodné jištění. Podle obecných pravidel je rozvaděč vybaven jističem u vstupu napájení do rozvodní skříně, kde se snižuje průřez vodiče, a dále před napájecím zdrojem jako ochrana proti jeho vnitřnímu zkratu. Za zdrojem pak stačí použít vhodné tavné pojistky. Nakonec by měla být jištěna i samotná zásuvka a to s ohledem na zařízení, která by do ní mohla být připojena.

Pro každý motor je potřeba v rozvaděči umístit měnič. Ty jsou dvojího typu. V anglické literatuře se setkáváme s pojmy „Motor driver“ a „Motor controller“, v obou případech se jedná o výkonovou součást, která zprostředkovává napájení a řízení motoru. V případě, že se jedná o Motor controller, většinou je vybaven také obvody určenými pro řízení a jeho logiku a řídicí signály získává přes komunikační rozhraní jako například RS232. V případě driveru je tato funkce přenechána jinému zařízení například MCU. V našem případě byly použity ovladače a řízení je součástí programu každého stroje jednotlivě a bude zmíněno v části věnující se programové výbavě.



Obrázek 10: Zapojení rozvaděče manipulátoru

Dále jsou v rozvaděči umístěny dva mikrokontroléry starající se o řízení procesu. Jeden z těchto mikrokontrolérů je zodpovědný za řízení polohovacího a druhý orientačního ústrojí. Tyto mikrokontroléry jsou doplněny komunikačním rozhraním CAN a 24V vstupně-výstupní logikou. Po otestování by měly být tyto komponenty integrovány na jednu desku umístěnou do krabičky s úchytem na DIN lištu. Veškeré vstupy i výstupy z rozvaděče jsou standardně



zajištěny průchodkami a svorkovnicemi s pružinovým mechanismem, jelikož pružinové kontakty se v praxi vyplatí nejen kvůli velmi rychlé montáži, ale také kvůli dlouhodobé odolnosti vůči vibracím. Dokumentace k rozvaděči je rozebrána v samostatné kapitole 7.

Z důvodů, které již byly rozvedeny v podkapitole 5.4), jsou všechny vodiče mimo rozvaděč vedeny stíněné a v párech kroucené, stínění je vždy na straně rozvaděče uzeměno.

## 5.6 Řídicí jednotka manipulátoru

Pro řízení úlohy byly použity dva mikrokontroléry vyráběné společností Microchip. Konkrétně ATmega2560 jako hlavní MCU a ATmega328p, která ovládá polohovacího ústrojí, tedy os XYZ. Ve fázi prototypování a testování je možné s výhodou využít těchto MCU vestavěných na platformách Arduino. Ty však pro praktické využití v průmyslu nejsou vhodné, pro jejich k tomuto účelu nepřizpůsobený návrh periferních obvodů a jejich nízké spolehlivosti. I tyto desky musely být vybaveny minimálně rozšiřující deskou se šroubovacími svorkami, aby mohly být použity jehlové kontakty, které jsou na deskách použity výrobcem totiž nejsou dostatečně robustní.

I po tomto zlepšení se však výstupní elektronika Arduino pro praxi nehodí, proto budou oba mikrokontroléry později nahrazeny vestavěným systémem vlastního návrhu integrující tyto dva mikrokontroléry, stejně jako potřebná komunikační rozhraní a napájení. Tento postup je efektivní, jelikož oba mikrokontroléry nabízejí možnost programování v C++, nabízejí patřičný počet vstupních a výstupních pinů a existuje pro ně řada knihoven a firmwarů, zároveň jsou velice levné a dostupné. Nahrazení platformy Arduino vlastním vestavěným je díky tomu omezeno na vyměnění součástky v rozvaděči bez nutnosti jakýchkoli zásahů do programu.

Poněkud dražší alternativou k tomuto řešení, by bylo využití zařízení Controllino, které jsou také postaveny na čípech ATmega, splňují však nároky IEC-61131 a CE. Prakticky se tedy jedná o open-source PLC. Cena těchto zařízení je však přibližně srovnatelná s PLC Siemens LOGO!. Zdá se vhodné zde uvést, že na první pohled smysluplný přechod přímo na PLC LOGO!, by byl krajně nevýhodný kvůli nízkému počtu vstupů a výstupů, stejně jako potřebě vyvinout zcela novou programovou výbavu. Použití samotného Controllina se jeví jako zbytečně nákladné v porovnání s výhodami zcela optimalizovaného vestavěného systému uvedeného v předchozím odstavci.

Potřeba vývoje vlastní elektroniky a robustního programu v C++, by v praxi mohla být považována za nevýhodnou vzhledem k časové náročnosti takového úkonu. V rámci akademické práce jako je tato, je však pro takový vývoj dostatek prostoru. Zároveň je takové řešení méně nákladné než typicky využívaná PLC, jako SIEMENS S7-1200, které by svou cenou několikanásobně zvyšovaly náklady na pořízení celého zařízení. Je potřeba vzít také v úvahu, že se nejedná o kritický systém, jelikož žádný typ poruchy zde nemůže způsobit zranění operátora, který se nachází mimo operační prostor manipulátoru. Kvůli nízkému výkonu motorů nemůže dojít ani k závažnému poškození okolních strojů.

## 5.7 Hlavní řídicí jednotka

Do procesu byl současně s pracemi na manipulátoru přidán mikrokontrolér zajišťující koordinaci všech strojů. Celá struktura je tedy založena na master-slave principu. Hlavními úkoly tohoto zařízení jsou komunikace s ovládacím panelem (podrobněji v kapitole 5.9), čtení signálů z univerzálních tlačítkových ovladačů a především - synchronizace činnosti strojů.

Aby tyto činnosti mohl co nejefektivněji vykonávat, je jeho software navržen jako stavový automat. Přesněji řečeno, v jednom cyklu obsluhuje několik stavových automatů, podle počtu najednou vyráběných dílů a během přechodů mezi jednotlivými stavy kontroluje komunikaci. Díky tomu je linka schopna nejen zdánlivého multitaskingu, ale zároveň rychlé odezvy při získávání a přeposílávání informací ze strojů na ovládací panel.

Každý výrobní proces má svůj vlastní stavový automat, který může být volán paralelně s jinými výrobními procesy. Samotný stav v praxi obsahuje jen soubor příkazů pro sadu vybraných strojů a podmínku přechodu z předchozího stavu. Pokud tato podmínka není splněna v patřičném čase, automat čeká tak, že v mezičase obsluhuje ostatní procesy. Díky tomu, že zpět na uživatelský panel zasílá informaci s číslem posledního stavu, je případná chyba snadno operátorem odhalena. Tato koncepce je s výhodou silně inspirována způsobem programování PLC.

## 5.8 Programové vybavení

Pro řízení polohování v osách XYZ, byl použit firmware GRBL. Ten je vyvinut pro CNC řízení rozličných zařízení, od obráběcích fréz přes 3D tiskárny až po laserové routery. Tento firmware je po více než 10 letech vývoje dobře optimalizován a zaručuje velmi plynulý a spolehlivý pohyb, umožňující nelineární pohyby, stejně jako plynulé zrychlení a zpomalení ve více osách najednou. Tento firmware obsahuje překladač G-kódu, díky čemuž je možné velice rychle upravovat program polohování zejména v prototypovací a testovací fázi. Tyto příkazy jsou přijímány přes sériovou linku, stejně jako speciální sekvence znaků určené pro parametrizaci překladače a další změny nastavení.

Komunikace s ostatními stroji, stejně jako řízení orientačního ústrojí a ukládání všech potřebných nastavení je pak zajištěna druhým MCU. Toto „hlavní“ MCU se také chová jako uživatel GRBL, obsahuje tedy programy pro řízení a vhodným způsobem odesílá tyto příkazy po seriové lince do sousedního „polohovacího“ MCU.

Každý příkaz G-kódu je v praxi textový řetězec, resp. v jazyce C se jedná o konstantní pole znaků. Soubory takových příkazů čítající stovky řádků jsou tedy náročné na paměť a nebylo by vhodné ukládat je jako proměnné. Navíc v této aplikaci jsou všechny příkazy polohování předem známé a mění se pouze při změně celého programu, proto se nabízí jako vhodné řešení přesunout je do jiné části paměti. Jednou možností je jejich uložení do programové paměti. K tomu se dá využít makro PROGMEM, dostupné ve standardní knihovně avr-libc. Druhou možností je přesunout je na externí paměť například SD kartu. Vzhledem k tomu, že po uvedení do provozu

již častější změny programu nebudou potřeba, jeví se jako vhodnější první možnost. Kód pro uložení do programové paměti pak můžeme napsat přímo podle doporučení v dokumentaci avr-libc.

Osy orientačního ústrojí jsou ovládány hlavním MCU. K ovládání motorů je samozřejmě dostupná řada knihoven, zejména pro usnadnění plynulých rozběhů je výhodné implementovat kupříkladu knihovnu AccelStepper, stejně jako GRBL šířenou pod licencí GPLv3. Jako alternativa byla současně napsána sada jednoduchých funkcí vysílajících periodické impulsy na výstupní piny spojené s příslušným měničem a umožňuje tak lineární pohyb a „homing“.

Rozhraní, které se shoduje s ostatními stroji funguje následovně. Program periodicky kontroluje komunikaci, odesílá své souřadnice, a stav a zároveň očekává příchozí příkazy. Tyto příkazy skladuje v zásobníku, dokud nejsou postupně vykonány všechny. Struktura komunikace je vysvětlena ve vlastní podkapitole. Pokud kterákoli z funkcí plní tyto příkazy některým z vnitřních kontrolních mechanismů detekuje chybu, přejde celý program do čekací smyčky a periodicky odesílá hlášení ostatním strojům a ovládacímu panelu.

## 5.9 Komunikace

Spolehlivá komunikace je nezbytná pro bezpečné řízení. Ztráta datového balíku během výroby by téměř jistě vedla k havárii. Zároveň schopnost odesílat na uživatelský panel data o stavu stroje je zásadní pro ovládání stroje, zejména v případě chybových stavů.

### Mezi mikrokontroléry manipulátoru

Oba mikrokontroléry mezi sebou komunikují přes UART na rychlosti 115200 baudů, s délkou zprávy 8 bitů, bez parity s 1 stop-bitem. Délka přenosového vedení je pouze několik centimetrů, nepočítá se tedy s větším indukovaným ani kapacitním rušením. Nerobustní komunikace bez přidáných kontrol může být příležitostí k chybě. Bylo však nutné vyhovět rozhraní GRBL a testováním se ukázalo, že zde nedochází k častějšímu výskytu chyb.

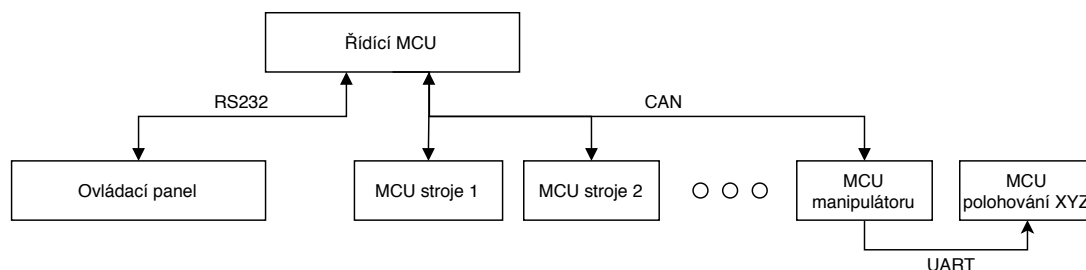
Aby mikrokontrolér řídící polohování fungoval správně, je potřeba zajistit, aby nebyl přeplněn jeho vstupní buffer, což je zajištěno počítáním odeslaných znaků a odečítáním potvrzení přijatých řetězců. Pro toto byla naprogramována funkce a to jak pro odesílání bloku kódu po částech, tak samostatných řádků kódu.

### S ostatními stroji

manipulátor komunikuje pomocí sběrnice CAN. Přesněji řečeno převodníku MCP2515 (Microchip), se kterým je komunikováno přes rozhraní SPI a vysokorychlostního vysílače/příjmače TJA1050 (Phillips). Oba tyto obvody jsou integrovány do kompaktního modulu.

Pro komunikaci byl využit standardní rámec obsahující 8 bytů, rychlost 250kb/s, příchozí data jsou vyčítána přes přerušení. Aby nedocházelo ke zbytečnému vyčítání zpráv, které nejsou pro manipulátor určeny, byly využity masky a filtry poskytnuté převodníkem MCP2515.

Komunikaci s ostatními stroji v praxi zastává mikrokontrolér zajišťující jejich synchronizaci. Ten získává data z manipulátorů a přeposílá je dále.



Obrázek 11: Umístění ovládacího rozhraní z pohledu komunikace

## 5.10 Datové rámce

Pro usnadnění čtení zpráv, každá zpráva obsahuje zvláštní znak pro označení začátku a konce zprávy. Tyto znaky odpovídající ASCII hodnotám 42 tj. '\*' a 33 tj. '!', jsou v rámci komunikace unikátní a je zajištěno, že se nemohou vyskytovat uvnitř zprávy. Ve výpisu 2 je tento přepočet implementován ve dvou funkcích odesílající data na sériovou linku. Každý byte informace je upravován tak, aby každá číselná hodnota byla přepočítána na hexadecimální v rozsahu 0x00–0xFF. Tato hodnota je pak rozdělena na dva digity a každému z nich je přiřazena odpovídající dekadická hodnota v ASCII tabulce. Pokud je numerická hodnota k odeslání větší než jeden byte, tedy 0xFF například z typu Int16, je nejdříve rozdělena (maskována) na dvě části a zpracována samostatně. Například tedy zpráva obsahující hodnotu 1 se převede na 0x01, tím pádem na písmena '0' a '1', a odešle se 48 a 49, což jsou odpovídající dekadické hodnoty podle ASCII tabulky. Tento proces zajistí, že všechny hodnoty jsou postupně převedeny do zúženého rozsahu dovolujícího snadnou kontrolu porušení zprávy. Kontrola integrity zpráv je navíc kontrolována koncovým součtem.<sup>1</sup>

V programech existují dva typy zprávy, a to podle směru, kterým jsou zasílány. Od operátora směrem ke strojům jsou zasílány příkazy, od strojů k operátorovi informace. Podle toho musí mít zprávy odlišný formát.

Formát zprávy směrem od operátora ke stroji je navržen tak, aby v co nejmenším počtu bytů bylo možné přenést veškeré informace potřebné k zadání libovolné funkce na kterémkoli ze strojů. Skládá se z několika částí, které jsou pro vyšší přehlednost pojmenovány následujícím způsobem „Adress, Command, Value, Flag“. Části obsahují v uvedeném pořadí: číslo cílového stroje; číslo příkazu, který se má vykonat; případnou hodnotu, která má být tímto příkazem nabyta; až 8 indikátorů typu boolean. Tyto části mají velikost jeden byte, s výjimkou Value, který má 16 bitů.

<sup>1</sup>Ve snaze o usnadnění pochopitelnosti textu, byly znaky ve významu různě reprezentovaných dat formátovány dle programátorských zvyklostí. Jednoduché uvozovky " tedy označují znakový literál. A znaky 0x označují počátek čísla v hexadecimálním tvaru.

Druhý typ zprávy je určen pro zasílání informací zpátky do řídicího MCU a k operátorovi. Pokud probíhá pouze výměna dat mezi mikrokontroléry, neodesílá se kontrolní součet, ten je přidán jen při přeposlání dat po seriové lince na ovládací panel. Zpráva má vždy maximálně 8B, což je na CAN sběrnici maximum standardní velikosti rámce. Tato zpráva je opět rozdělena na pojmenované části „Adress, State, Queue, Homed, Info, Info, Info, Info“. Tyto části udávají v uvedeném pořadí: adresu cílového stroje; boolean udávající, zda je stroj právě v pohybu; počet příkazů ve frontě k vykonání; informaci o tom, zda byl proveden homing; 4 hodnoty, které nemusí být využity, pokud nejsou potřeba.

---

```
void send(byte data[]) {
    byte sum = 0; //Koncový součet, přetečení je záměrné

    Serial.write(START); //Startovací znak '*'
    //Odešle data a zároveň počítá koncový součet
    for (int i = 0; i < DATA_LENGTH; i++) {
        sum += data[i];
        convertOut(data[i]);
    }
    convertOut(sum);
    Serial.write(STOP); //koncovy znak '!'
}

void convertOut(uint8_t)
{
    //Rozdělení na dva digity A a B
    char partB = data & 0x0F;
    char partA = (data & 0xF0) >> 4;
    //Přepočet v rozsahu ASCII znaků
    partA < 0x0A ? partA += 0x30 : partA += 0x37;
    partB < 0x0A ? partB += 0x30 : partB += 0x37;
    Serial.write(maskA);
    Serial.write(maskB);
}
```

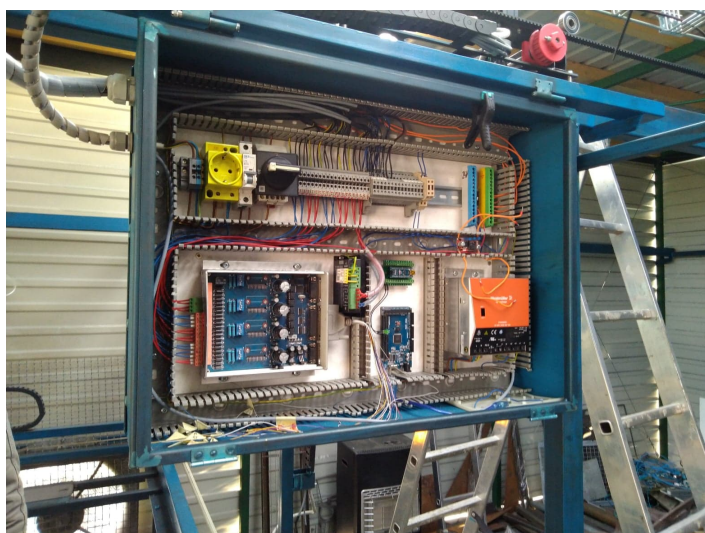
---

Výpis 2: Ukázka funkcí pro zasílání dat po sériové lince

## 6 Postup sestavení

Manipulátor vznikal postupně v několika fázích nejprve bylo potřeba sestavit jeho mechanickou konstrukci. Z již mnohokrát zmíněného požadavku na minimalizaci výrobních nákladů, byla naprostá většina dílů vytvořena ručně, takzvaně „in-house“. K tomu bylo možné využít také některé ze strojů výrobní linky, jako je plasmové řezání. V případě těžko zakoupitelných dílů, nebo dílů s atypickými rozměry, bylo také využito 3D tisku na tiskárně Průša i3. Mezi takové díly se řadí například kryty kladek osy Z. Na podobrázku 12b je zobrazena první sestavená verze manipulátoru, bez vylepšení popsanych na konci této kapitoly.

Dalším krokem bylo vytvoření rozvaděče a připojení všech potřebných vodičů. Proces montáže je zachycen na podobrázku 12a. Již zapojený rozvaděč na obrázku 10 je, přestože zcela funkční, považován za přechodnou verzi. Dalším krokem by mělo být vybavení ovladači pro zpětnovazební řízení motorů, novými MCU a DC/DC měnič bude buď nahrazen druhým zdrojem, nebo alespoň umístěn do krabičky a připojen na DIN lištu.



(a) Průběh práce na rozvaděči



(b) Manipulátor před přidáním magnetů

Obrázek 12: Rozvaděč v průběhu montáže

Poté byly řešeny problémy s rušením způsobeným zápalom plasmového hořáku, zmíněné v podkapitole 5.4. V této fázi výroby došlo k výraznému zpomalení, jelikož problém se netýkal pouze manipulátoru. Zásadní bylo například chybné odečítání enkodéru hydraulického lisu měřícího zdvih. Významná porucha tohoto snímače může mít za následek zničení celého stroje, v lepším případě pouze v daný moment vyráběného dílu. K těmto chybám v důsledku rušení několikrát došlo, a proto bylo potřeba tento problém odstranit. Na obrázku 13 vlevo je obrazovka z osciloskopu zachycující zarušený signál enkodéru. Na stejném obrázku v pravo je uveden signál měřený za ideálních podmínek, bez přidaného rušení frekvenčních měničů a zážehů plasmy.

I na druhém obrázku je vidět, že průběh není zcela ideální. Tyto průběhy je však nezbytné vyčítat správně za provozních podmínek, toho bylo dosaženo za pomoci postupů uvedených v podkapitole 5.4.



(a) Signál s výrazným rušením

(b) Signál po základní filtraci

Obrázek 13: Zobrazení různých intenzit rušení na osciloskopu

Nakonec byl řešen samotný software v této fázi již bylo vytvořeno jedno uživatelské rozhraní v C# zmíněné v kapitole 8. Nejprve byl vytvořen program hlavního řídicího MCU výrobní linky, který umožňoval zaslání pokynů k výrobě celého dílu do libovolného stroje. Poté byl pareametrován a konfigurován firmware GRBL. Nakonec byl vytvořen program ovládající zbylé dvě osy a zprostředkovávající komunikaci. Vzhledem k možnosti zpětně získávat z GRBL souřadnice manipulátoru, je možné manipulátor snadno programovat následujícím postupem. Operátor ručním ovládáním nalezne a odečte souřadnice, programátor je vloží do hlavního programu mikrokontroléru. Pochopitelně by nebylo obtížné umožnit manipulátoru pružné programování z operátorského panelu, vzhledem k jednoúčelové povaze výroby by však taková funkce v praktickém provozu neměla žádné využití.

Po uvedení do provozu a základním testování bylo zjištěno, že po mechanické stránce je potřeba provést několik úprav. Mezi nimi zejména vytvoření krytů kladek osy Z, jelikož při povolení lanka, ke kterému docházelo při přílišném spuštění nad uchopovaný objekt, docházelo k seskočení lanka s kladky. Dále pak počet magnetů koncového efektoru musel být zvýšen ze dvou na čtyři, poněvadž zejména dna a boky po lisování nebyly zcela rovné a nižší kontaktní plocha dostatečně snižovala přichytnou sílu, aby díly při převozu odpadly.

## 7 Dokumentace

Aby bylo možné manipulátor sestavit, provozovat a udržovat, je potřeba vytvořit jeho dokumentaci. Sestrojení mechanického řešení není předmětem této práce, proto zde nespadá ani tato část dokumentace. Technická dokumentace v takovém případě pak obnáší zejména elektro-technickou dokumentaci rozvaděče a jeho vnitřního zapojení.

### 7.1 Eplan

EPLAN Electric P8 je softwarem stejnojmenné společnosti (skupina Friedhelm Loh Group), umožňující projektování technické dokumentace. Je možné jej zařadit pod programy typu CAE, čili software usnadňující inženýrskou analýzu. Podporuje různé mezinárodní normy jako IEC, NFPA, ruské GOST nebo čínské GB s příslušnými kmenovými daty a vzorovými projekty. [16] Kromě Electric P8, společnost Eplan nabízí další užitečné nástroje jako je Pro Panel a mnohé další.

### 7.2 Zapojení rozvaděče

Projektování rozvaděče je pochopitelně další samostatný obor, kterého se tato práce dotýká, proto bylo nezbytné jej v dostatečné míře konzultovat s projektanty z praxe.

Hlavní částí dokumentace rozvaděče je vícepólové schéma zahrnující na několika stranách napájení a jistění, dále pohony a aktuátory a nakonec připojení MCU. Tato část je uvedena v příloze B.

**Část napájení** – předpokládá se, že mezi přípojným bodem rozvodné sítě a rozvaděčem manipulátoru je ještě jedna hlavní rozvodná skříň s pojistkami, jistící celou výrobní halu. Jelikož je rozvaděč umístěn dále od ostatních pracovišť, je rozvaděč vybaven vlastní zásuvkou 230 V, která se hodila zejména ve fázi vývoje pro zapojení zařízení užívaných k montáži nebo opravám.

**Část pohonů** – naznačuje zapojení všech krokových motorů na příslušné měniče, v případě zavedení zpětnovazebně řízených motorů, by každý motor disponoval vlastním měničem a bylo nutno dokumentaci přepracovat. Zároveň je zde uvedeno zapojení cívek elektromagnetů koncového efektoru. Všechny vodiče mimo rozvaděč jsou vedeny stíněnými kabely aby bylo omezeno elektromagnetické rušení.

**Část připojení MCU** – předpokládá využití MCU Atmega, jejich zapojení se v případě, výměny prototypovacích desek za finální návrhy nemění, proto zde dokumentace slouží jako návod k zapojení.



## 8 Uživatelské rozhraní

Při používání linky v každodenním provozu, je pro její obsluhu nejdůležitější ovládací rozhraní. Pokud je přehledné a snadno ovladatelné, je práce mnohem efektivnější z pohledu kapacity výroby a je také mnohem snazší naučit se s ním pracovat. Z toho důvodu byl kladen důraz na jeho dobré grafické zpracování a komunikaci s budoucím operátorem.

Větší část ovládání je zprostředkována za pomoci programu v PC, který obsahuje uživatelské rozhraní. Program se chová jako periférie, která s pomocí seriové komunikace zasílá do řídicího MCU příkazy od uživatele. Tyto příkazy jsou následně rozeslány do MCU řídicích jednotlivé stroje, kde jsou vykonány. Tuto topologii naznačuje schéma na obrázku 11.

Protože ovládací rozhraní neobsahuje žádnou logiku výsledného řízení, můžeme implementovat více než jedno ovládací rozhraní a tím dále usnadnit ovládání při každodenní práci.

### 8.1 Hardwarové prostředky

Mezi tyto možnosti se řadí tyto prostředky:

- Tlačítkové a indikátorové panely.
- Vestavěné operátorské panely s dotykovým displejem nebo tlačítka.
- Dálkové ovladače s tlačítka.
- Počítačové pracoviště s vhodným softwarovým vybavením.



Obrázek 14: Ovladač hydraulického lisu

Jak již bylo uvedeno výše, základem uživatelského rozhraní je právě počítačové pracoviště a to z těchto důvodů. Díky dostatečné paměti a výkonu dovoluje snadnou rozšiřitelnost o funkce jako je přenos obrazu z kamer, které významně zlepšují operátorův přehled o dění na pracovišti. Nebo statistiku dat nasbíraných při výrobě.

Další možností je využít například jednodeskový počítač Raspberry PI a vyměnit desktopové rozvržení za vestavěný ovládací panel s dotykovým displejem. Uživatelské rozhraní může být umístěno ve vhodné vzdálenosti od zařízení, čímž je pasivně ochráněno před nepříznivými vlivy okolí jako EMI, prach nebo vibrace a nevznikají tak nároky na jeho odolnost. V tomto uspořádání se nabízí četné softwarové možnosti, uvedené v seznamu níže.

U některých pracovišť je však velmi nepraktické a časově nákladné, neustále přecházet mezi pracovištěm a PC/HMI panelem, pouze kvůli zapnutí dalšího cyklu výroby na ovládacím panelu. Proto bylo k zařízení přidáno několik mechanických ovladačů, které umožňují spouštět a zastavovat některé specifické úlohy.

Úlohy, jako například lisování, vyžadují kvůli bezpečnosti speciální typy spínačů, jako je na obrázku 14, vyžadující použití obou končetin, aby bylo s jistotou zamezeno nesprávnému zacházení. U ostatních ovladačů je však možné jejich funkce programově měnit podle aktuální potřeby a tím je snížen počet potřebných hardwarových komponent. To má pozitivní dopad na výslednou cenu zařízení.

## 8.2 Softwarové prostředky

Mezi ty se řadí tyto prostředky:

- C/Python + GTK, je multiplatformní, licence GNU
- C# + .NET, nabízí UI známý většině uživatelů
- LabView VI, umožňuje rychlý a snadný vývoj
- HTML + JavaScript, online aplikace usnadní dálkový přístup, pokud je potřeba.

Z těchto byly pro realizaci vybrány dvě možnosti, C# a LabView, z nichž každá je popsána ve vlastní podkapitole. C# byl vybrán z toho důvodu, že v něm již byla napsána předchozí verze a k dispozici byl rozsáhlý kód, na nějž bylo možné navázat. LabView pak zejména pro srovnání výhod obou přístupů a případný přechod na tuto platformu. V případě, že by měl být přidán ovládací panel s jiným OS než Windows, například právě Raspberry PI, program v C# by nebyl kompatibilní, pro LabView existuje dokoupitelný kompilátor, avšak nejlepšího výsledku by bylo dosaženo napsáním zcela nového programu například pomocí C++, případně Pythonu, a GTK.

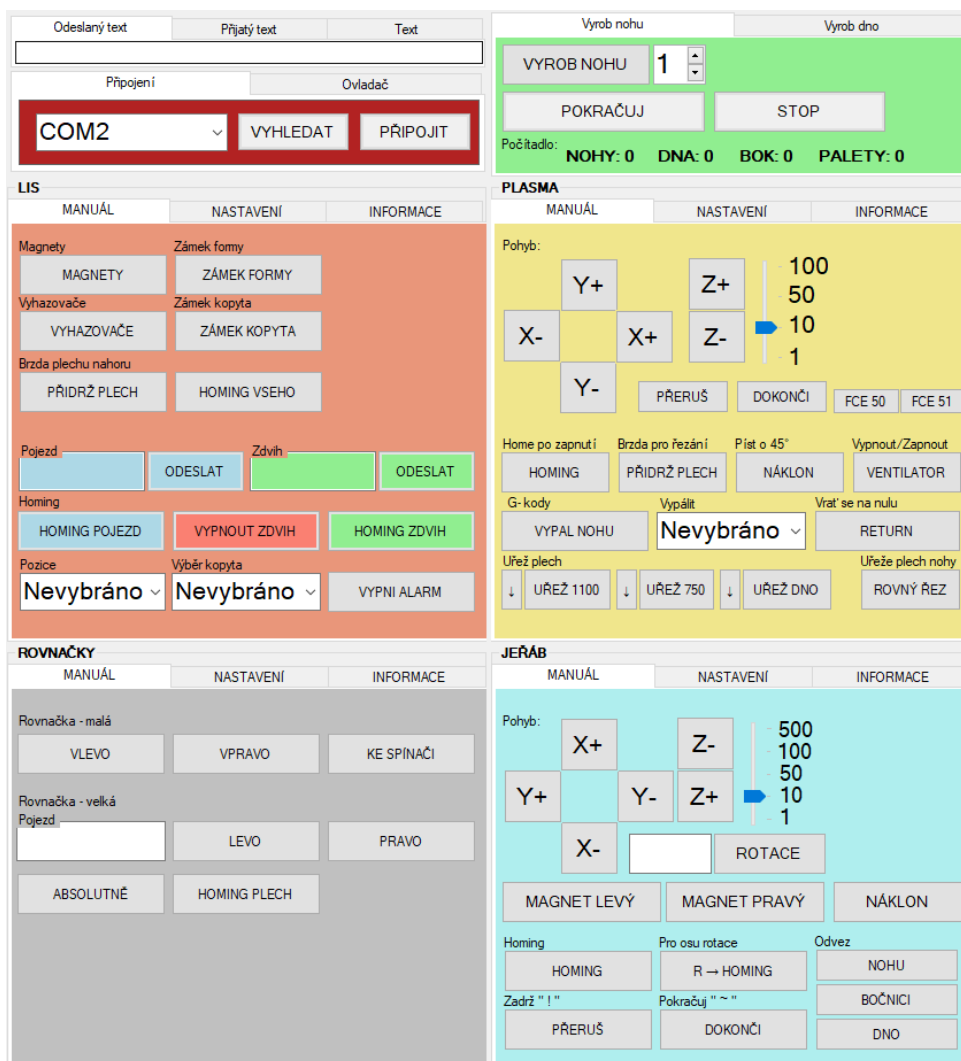
## 8.3 Návrh rozhraní

Z pohledu obsluhy je program rozdělen do samostatných ,barevně odlišených karet. Ty odpovídají buď strojům zařazeným do výrobní linky - karty manuálního ovládání, nebo výrobnímu procesu, například rovnání plechu nebo výroba dílu - karty automatického řízení. Ty pak obsahují všechny potřebné ovládací prvky. Karty, které jsou určeny k manuálnímu ovládání pak navíc obsahují část s ukládáním uživatelských nastavení a část poskytující informace o stavu stroje uživateli. Dále se zde nacházejí komponenty pro ovládání připojení k lince, kontrolu příchozích a odchozích dat a další užitečné nástroje.

Z pohledu kódu v pozadí je hlavní úlohou programu skládání zpráv na základně uživatelského vstupu tak, aby odpovídaly předem zvolenému komunikačnímu protokolu, nebo naopak z příchozích zpráv zobrazil příslušné upozornění, nebo chybové hlášení.

## 8.4 Rozhraní v C#

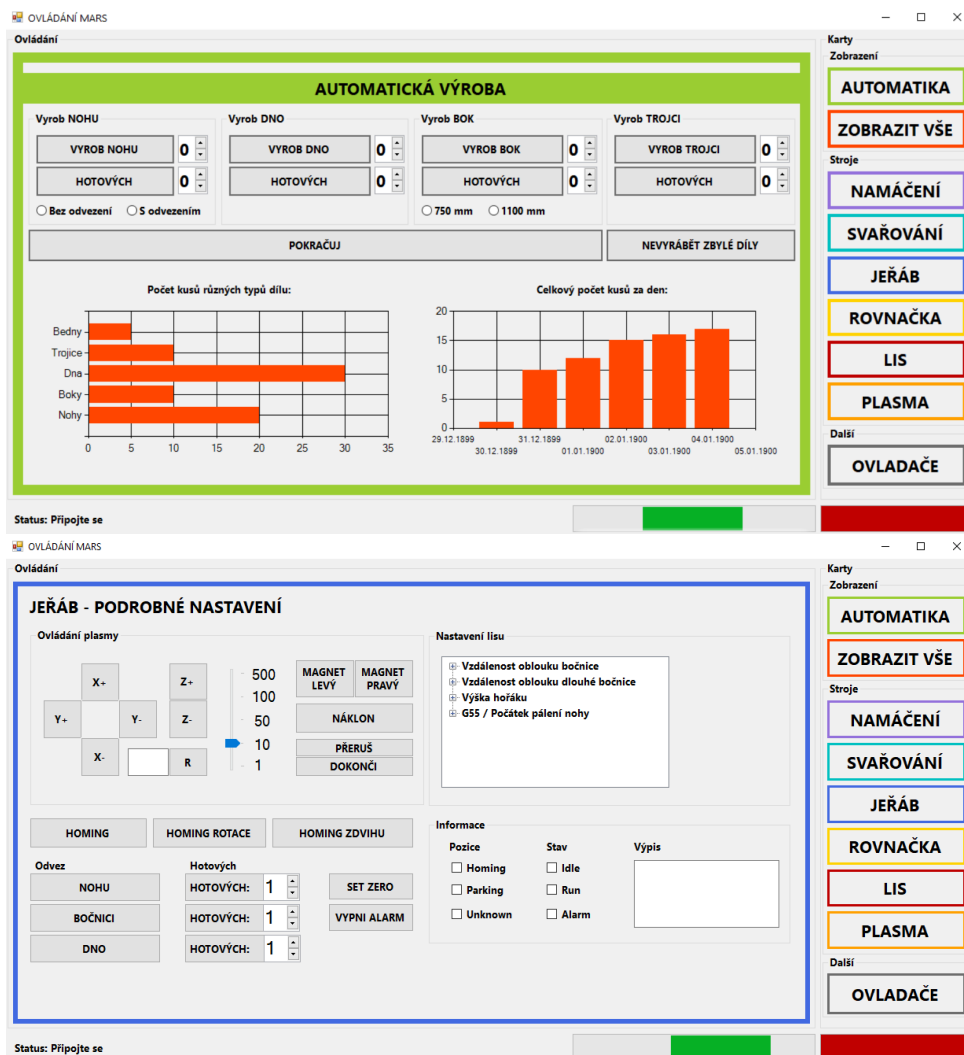
Už před začátkem práce byl k dispozici rozpracovaný program, vytvořený pomocí nástrojů Windows Forms, tento je zobrazen, na obrázku 15, fungovaly v něm však pouze základní funkce manuálního ovládání. Připojování probíhalo ručně a z praktických zkušeností začínalo být zřejmé, že kvůli množství ovládacích prvků přestává být přehledné.<sup>2</sup>



Obrázek 15: Staré rozhraní v C#

Proto bylo přepracováno tak, aby tyto nedostatky byly opraveny. Výsledek je zobrazen na obrázku 16, na pravé straně byla přidána lišta s nabídkou zobrazených strojů, ta se hodí zejména v případě použití dotykového displeje, jelikož umožňuje vytvoření karty s libovolným počtem ovládacích prvků různých velikostí.

<sup>2</sup>Obrázky uživatelských rozhraní ve vyšší kvalitě jsou uvedeny v příloze C.



Obrázek 16: Nové rozhraní v C#

Zároveň byl dán větší prostor kartě automatického řízení, která díky zdokonalení řídicího programu bývá využívána častěji než manuální ovládání. Je teď navíc snadněji dostupná hned po zapnutí.

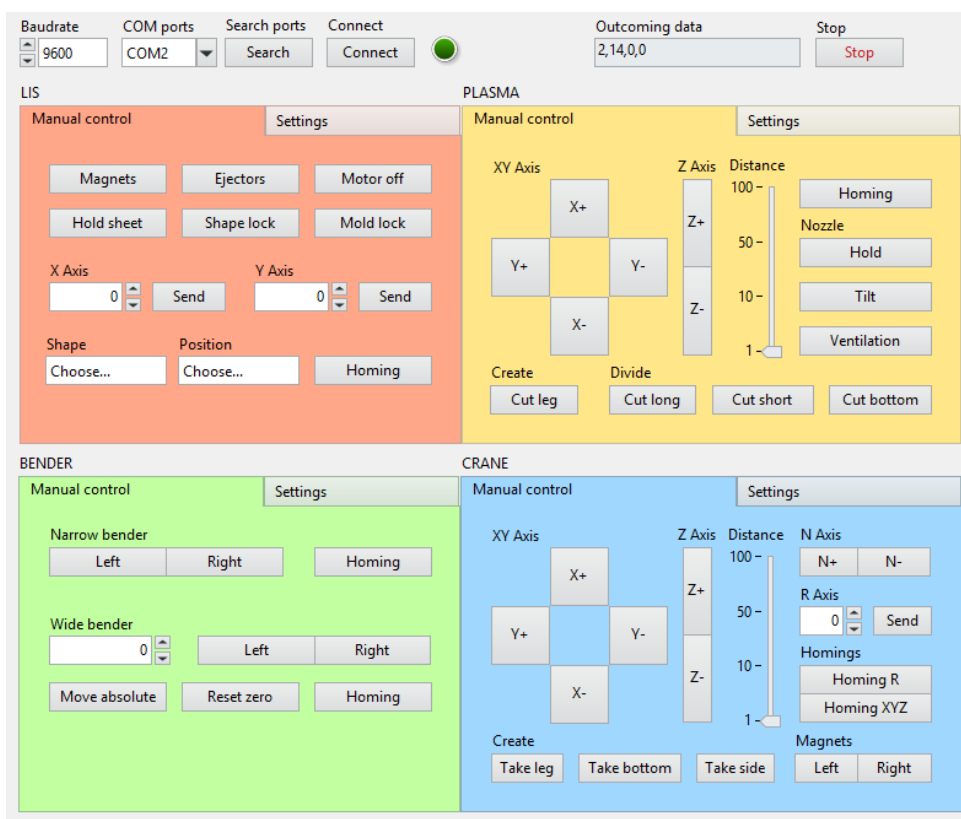
Zároveň byla odstraněna nabídka s připojením a komunikací, protože tyto funkce byly automatizovány a probíhají bez účasti operátora. V případě potřeby jsou dostupné na samostatném panelu zapnutelném v nastavení.

Nakonec bylo přidáno několik užitečných funkcí jako stavová lišta v dolní části, vyvolatelný poznámkový blok, nebo počítadlo hotových dílů se statistikou v grafech.

Kód v pozadí byl vytvořen také nový, lépe strukturovaný z pohledu objektově orientovaného návrhu aplikace. Komunikace byla zásadně rozšířena tak, aby podporovala i příjem dat z řídicího MCU a tím byla schopná zobrazovat stav stroje. Dále byla zavedena podpora místního ukládání parametrů nastavení strojů do souborů XML.

## 8.5 Rozhraní v LabView

Rozhraní v LabView bylo vytvořeno podle staršího rozhraní zejména za účelem srovnání, případně jako alternativa. Pokud by bylo uživatelské rozhraní v budoucnu přesunuto na linuxovou platformu jako je Raspberry PI, pomocí kompilátoru lze VI přeložit do podoby spustitelné i na Raspbianu. Hlavní výhodou LabView je bezesporu rychlý vývoj nových aplikací, také proto jeho obliba v praxi poslední dobou stále roste.



Obrázek 17: Nové rozhraní v LabView

Návrh kódu je zde velice rychlý, obzvláště při použití vhodného návrhového vzoru jako je „Producer-Consumer design pattern“. LabView je svou koncepcí více podobné vysoko-úrovňovým jazykům jako Python nebo C# , a nízkoúrovňové úpravy jako složitější manipulace s bity či pamětí jsou nejspíše tak nativní.

V ostatních ohledech je však patrné, že LabView je velice silným nástrojem nejen rapid-prototypingu. Jelikož u aplikací, jako je jednoduchý ovládací panel, většinou není potřeba tolik dbát na efektivní správu paměti nebo bitové manipulace.

Na místech, kde byly využity standardní postupy a komponenty LV, je sice tento způsob patrně efektivnější, na druhé straně tam, kde vznikl požadavek na vytvoření vlastní komponenty (náhrada jednoho z bloků VISA), bylo programování v LV časově i náročnostně srovnatelné s jeho alternativami.

## 9 Sběr dat a testování

Poté, co byl manipulátor sestaven a naprogramován, bylo zapotřebí jej vhodným způsobem otestovat tak, aby mohl být předán operátorovi s jistotou, že nebude docházet k častým poruchám.

### 9.1 Testování

Vzhledem k tomu, že mnohé použité komponenty byly recyklovány ze starších strojů a až na několik výjimek (ovladače, elektromagnety, snímače) ručně vyrobeny, bylo nejprve potřeba otestovat, že všechny pohyblivé součásti jsou schopné pohybu v celém jejich rozsahu. K tomu byly nejprve naprogramovány cyklické programy vykonávající jeden či dva izolované pohyby, například vypnutí - zapnutí náklonu. Tyto programy byly spuštěny v pěti stech opakováních, čímž byla zkontrolována jejich funkčnost a zároveň zajištěna dostatečná distribuce maziva.

Následující test obnášel odvážení výrobku ručním ovládáním, zatímco operátor zapisoval jeho souřadnice, aby mohly být později automaticky vykonávány programem řídicího MCU. V tomto režimu manipulátor pracoval několik cyklů, aby programátor s operátorem mohli ověřit vhodnost úložných pozic, přesnost pohybu a další parametry.

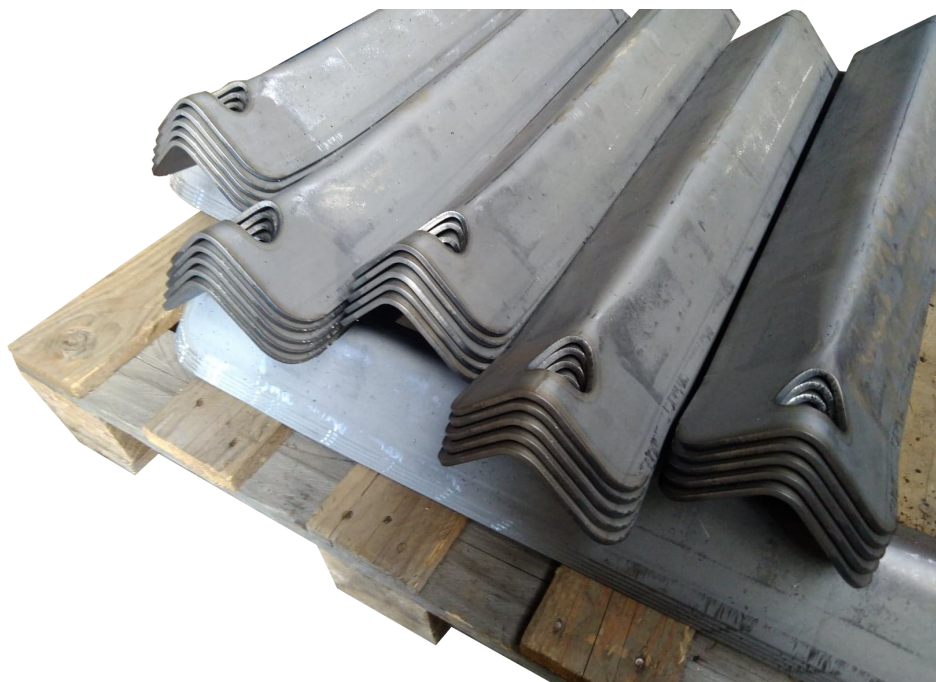
Dalším krokem bylo otestovat manipulátor v omezeném rozsahu, a to pouze pro odvážení výrobků z pracoviště. V tomto režimu několik týdnů odebíral manipulátor nohy z routeru plasmu a ukládal je na odkládací paletu uloženou poblíž viz obrázek 18. Takovým způsobem bylo zajištěno, že manipulátor nezpůsobí havárii zamrznutím na nevhodném místě, jako například u posuvu hydraulického lisu. Takovým způsobem bylo v testovacím režimu uloženo a zpracováno několik set kusů.

Posledním stupněm testování bylo zavedení plně automatického pohybu manipulátoru při výrobě. V tomto případě se již manipulátor pohybuje autonomně (ve smyslu bez nutnosti obsluhy operátorem, stále však podle pevného programu), zároveň je však snížena jeho pojezdová rychlost, aby v případě poruchy nebo chyby programu měl operátor dost času stroj zastavit, například tlačítkem nouzového vypnutí.

### 9.2 Ukládání dat

Dle zadání má fáze testování a měření implementovat vybrané prvky Industry 4.0. Pod tímto pojmem, ministerstvem průmyslu a obchodu překládaným jako Průmysl 4.0, můžeme rozumět celou filosofii přechodu k automatizovanější, modernější společnosti. [17][18] Proto se zdá být nutné specifikovat, jakým způsobem bude tato filosofie implementována.

Jedna z klíčových vizí Průmyslu 4.0, je tzv. „Vertikální intergrace všech subsystémů“, čímž je myšleno propojení od nejnižší úrovně automatizace fyzických procesů až po ERP, čili plánování podnikových zdrojů. Takové propojení je viditelné například na obrázku 16. Ze statistiky vyrobených dílů můžeme díky znalosti rozměrů každého výrobku snadno vypočítat spotřebu



Obrázek 18: Rohové díly na odkládací paletě

materiálu a to včetně materiálu svařovacího. Díky tomu může rozhraní snadno upozornit operátora na nutnost objednávky nového materiálu. Můžeme dokonce jít o krok dál a při vzniku takového hlášení přímo odeslat emailem poptávku dodavateli.

Dalším strojem, který v rámci automatizace můžeme takto vylepšit zavedením myšlenek Průmyslu 4.0 je router plasmového řezání. Tryska průmyslového hořáku v něm použitá je spotřebním dílem, který je přibližně každých 100 nařezaných dílů nutné vyměnit. V případě výměny, která není očekávána, musí operátor stroj a výrobu zastavit, aby mohl vstoupit do pracovního prostoru. Poté vymění trysku a celý stroj znovu uvede do provozu. Tato výměna může zabrat až 15 minut a opakuje se poměrně často (v závislosti na vlhkosti vzduchu). Počítáním vyrobených kusů případně, nebo lépe, počtu zápalů, můžeme vytvořit statistiku, díky níž můžeme lépe predikovat, kdy bude nutné trysku vyměnit. Predikcí nutnosti výměny může stroj kontrolovaně přijet k operátorovi s požadavkem na výměnu a čas nezbytný k provedení této operace se zkrátí na 2-5 minut, což z dlouhodobého hlediska může být významnou úsporou.

Například kontinuálním měřením tlaku můžeme lépe regulovat doplňování vzduchu kompresorem a předcházet tak zastavení práce v důsledku čekání na doplnění tlaku v pneumatických rozvodech. Prakticky všechny součásti použité v této práci jsou vyráběné takzvaně „in-house“, čili v dílně, kde vzniká i samotný manipulátor a často z recyklovaných dílů. Sledování životnosti součástí zde tedy velký význam nemá.

Nakonec všechna tato data je dobré mít možnost sledovat vzdáleně. Zejména informace o nutnosti doplňování materiálu a počtu vyrobených kusů by bylo dobré vzdáleně poskytnout

obchodnímu zástupci firmy, tak aby mohl s předstihem objednat nový materiál nebo věděl jak, velké může v daném okamžiku přijímat zakázky.

Za tímto účelem můžeme připojit celý manipulátor, případně automatizovaný řetězec strojů k internetu - cloudovému uložišti. Nejjednodušším řešením je připojit modul s možností přístupu k internetu, například desku NodeMCU s modulem ESP. Můžeme tak přijímat informace z řídicí jednotky a ukládat je na firemní server. Druhá možnost je využít Raspberry PI, což by se jevílo jako obzvláště vhodné v případě, že by byla implementována HMI obrazovka využívající tuto platformu. V takovém případě by proces ukládání dat online byl mnohem jednodušší, jelikož by bylo k dispozici i grafické rozhraní.



## 10 Závěr

Vypracováním této práce vznikl nejen plně fungující robotický manipulátor, ale také automatický výrobní proces, který umožňuje výrobu dílů ohradových palet. Tento proces sice stále vyžaduje operátora, aby funkci kontroloval, vyměňoval spotřební díly a zakládal nový materiál, samotná práce na výrobě dílů je však nyní zcela vykonávána strojem.

Přidáním manipulátoru byla zejména odstraněna potřeba přenášení dílů mezi pracovišti, která byla fyzicky a časově náročná. Rychlost výroby byla násobně zvýšena, dalšího zrychlení by bylo možné docílit výměnou nebo přidáním některých součástí. Například přidáním zpětné vazby krokovým motorům, což umožní bezpečné zvýšení přejezdových rychlostí. Nebo rozdělením úloh univerzálního hydraulického lisu na několik specializovanějších zařízení pro lisování různých dílů, které by nemusely tak často vyměňovat nástroje a fungovat souběžně na výrobě různých dílů.

Tato práce tématicky zasahuje do většiny oborů a problematik studovaných v rámci programu Řídicí a informační systémy, často rozsahem či obtížností přesahující probíranou tematiku bakalářského studia. Byly zpracovány projektové dokumentace, které budou vyžadovat schválení odborníky a při samotné výrobě bylo potřeba znalostí robotiky, mechatroniky, elektroniky a programování. Bohužel předměty, které byly pro získání znalostí nutných k vypracování práce nejvíce potřebné byly v rámci studia umístěny až do posledního semestru, proto větší část práce vznikla pouze na základě konzultací a studia odborné literatury.

Jak již bylo naznačeno v textu práce, manipulátor stále poskytuje prostor pro výrazné zlepšení. Zejména v rozvaděči by bylo, doplnit kvalitnější a spolehlivější ovladače motorů. Dále bude v budoucnu potřeba rozšířit program tak, aby zahrnoval práci s přibývajícimi pracovišti, jako je svařecí automat. V blízké době bude také postaven druhý podobný manipulátor s robustnější mechanikou, zajišťující automatizaci namáčení a skladování svařených beden. Tento nový robot bude v mnohém stavět na zkušenostech získaných při práci na tomto manipulátoru. V době jeho stavby by měly být také dostupné prostředky na pořízení kvalitnějších komponent.

Díky toho, že v rámci této práce byly lépe prozkoumány prostředky redukce rušivého signálu na vstupech strojů, bude v budoucnu možné tyto stroje vybavit více senzory. To samo o sobě zvýší kvalitu a efektivitu výroby a umožní zvýšení produktivity. Další krokem bude vytvoření svařovacího stanoviště a jeřábu automatizujícího namáčení a skladování beden.

## Literatura

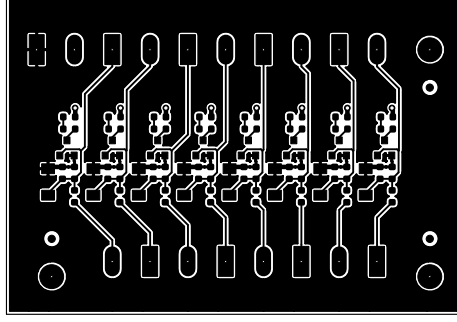
1. FLODROVÁ, Milena; LOSKOTOVÁ, Irena. *První protest dělníků proti zavádění strojů do výroby v Brně* [online]. Brno: Moravská zemská knihovna v Brně, 2005 [cit. 2020-03-04]. Dostupné z: [https://encyklopedie.brna.cz/home-mmb/?acc=profil\\_udalosti%5C&load=5413](https://encyklopedie.brna.cz/home-mmb/?acc=profil_udalosti%5C&load=5413).
2. *Defining The Industrial Robot: Industry and All It Entails* [online]. Michigan, USA: Robotic Industries Association, 2008 [cit. 2020-04-23]. Dostupné z: [www.robotics.org](http://www.robotics.org).
3. *Roboty a robotická zařízení: Slovník*. Praha: Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, 2013.
4. WAMPLER, Charles; SOMMESE, Andrew. *Numerical algebraic geometry and algebraic kinematics*. 2011-04. Dostupné z DOI: 10.1017/S0962492911000067.
5. CHVÁLA, Břetislav; MATIČKA, Robert; TALÁCKO, Jaroslav. *Průmyslové roboty a manipulátory*. 1. vydání. Spálení 51, 113 02 Praha 1: SNTL - Nakladatelství technické literatury, 1990. ISBN 80-03-00361-X.
6. SCHLEGEL, Miloš; ŠVEJDA, Martin; GOUBEJ, Martin. *Uvod do mechatroniky, robotiky a systémů řízení pohybu: Skriptum pro studenty doktorských programů v oboru automatického řízení*. 1. vydání. Plzeň: Západočeská univerzita v Plzni, 2012.
7. KERNIGHAN, Brian W.; RITCHIE, Dennis M. *Programovací jazyk C*. Brno: Computer Press, 2006. ISBN 80-251-0897-X.
8. *MISRA-C:2004: Guidelines for the use of the C language in critical systems*. 1. edition. Warwickshire, UK: MIRA Limited, Říjen 2004.
9. MONK, Simon. *Programming the Raspberry Pi<sup>TM</sup>: getting started with Python*. New York: McGraw-Hill, 2013. ISBN 978-0071807838.
10. VOJÁČEK, Antonín. *Magnetické senzory přiblížení - 2. díl* [online]. Formanská 296, 149 00 Praha 4: HW server s.r.o., 2014 [cit. 2018-12-08]. Dostupné z: <https://automatizace.hw.cz/magneticke-senzory-priblizeni-2-dil.html>.
11. BUDIMIR, Miles. *Hybrid servos vs. standard servos: How do they stack up?* [online]. USA: Miles, Budimir, 2017 [cit. 2020-04-27]. Dostupné z: [motioncontroltips.com](http://motioncontroltips.com).
12. SKAŘUPA, Jiří. *Průmyslové roboty a manipulátory*. Ostrava: Vysoká škola báňská - Technická univerzita, 2008. ISBN SBN978-80-248-1522-0.
13. VOJÁČEK, Antonín. *Nouzové zastavení stroje. Volba a použití tlačítek E-STOP* [online]. Formanská 296, 149 00 Praha 4: HW server s.r.o., 2014 [cit. 2020-04-27]. Dostupné z: <https://automatizace.hw.cz>.
14. HOROWITZ, Paul; HILL, Winfield. *The art of electronics*. Third edition. New York: Cambridge University Press, 2015. ISBN ISBN978-0-521-80926-9.

15. *Datasheet BS170: N-channel vertical D-MOS transistor* [online]. Praha: GME, 1990 [cit. 2020-05-07]. Dostupné z: <https://www.gme.cz/data/attachments/dsh.213-008.1.pdf>.
16. *EPLAN Electric P8: energie pro elektrotechnické projektování* [online]. Praha: Eplan, 2018 [cit. 2020-04-28]. Dostupné z: <https://www.eplan.cz/cz/reseni/elektrotechnika/eplan-electric-p8/>.
17. ODBOR, 31300. *Iniciativa Průmysl 4.0* [online]. Praha: MPO, 2016 [cit. 2020-04-28]. Dostupné z: <https://www.mpo.cz/cz/prumysl/zpracovatelsky-prumysl/prumysl-4-0-ma-v-cesku-sve-misto--176055/>.
18. VOJÁČEK, Antonín. *Co se skrývá pod výrazy Industry 4.0 / Průmysl 4.0 ?* [online]. Praha: HW server s.r.o., 2014 [cit. 2020-04-28]. Dostupné z: <https://automatizace.hw.cz/mimochodem/co-je-se-skryva-pod-vyrazy-industry-40-prumysl-40.html>.

## A Dokumentace DPS

V této příloze jsou uvedeny výkresy DPS úpravy signálu pro čtení digitálních vstupů. Na následujících stranách jsou uvedeny v tomto pořadí:

- Výkres DPS strana 1, vrstva „Top“
- Výkres DPS strana 2, vrstva „Bottom“



DPS určená ke zvýšení kvality čtení digitálních vstupů  
Petr Janoušek, Ing. Golembiovský  
Rozměry v měřítku 1:1, 60x41mm

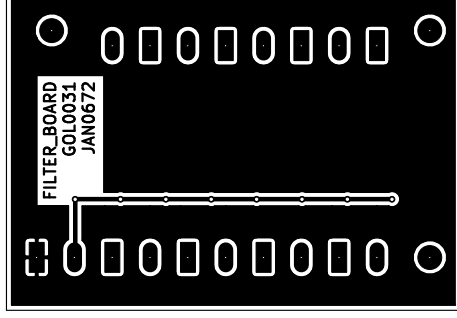
**VŠB-TUO FEI-KAT450**

Sheet:  
File: odruseni\_koncaku.kicad\_pcb

Title: **Bakalářská práce – FiltrDPSv2**

Size: A4	Date: 2020-05-01
----------	------------------

Id: 1/1	
KiCad E.D.A. pcbnew (5.0.2)-1	



DPS určená ke zvýšení kvality čtení digitálních vstupů  
Petr Janoušek, Ing. Golembiovský  
Rozměry v měřítku 1:1, 60x41mm

**VŠB-TUO FEI-KAT450**

Sheet:  
File: odruseni\_koncaku.kicad\_pcb

Title: **Bakalářská práce – FiltrDPSv2**

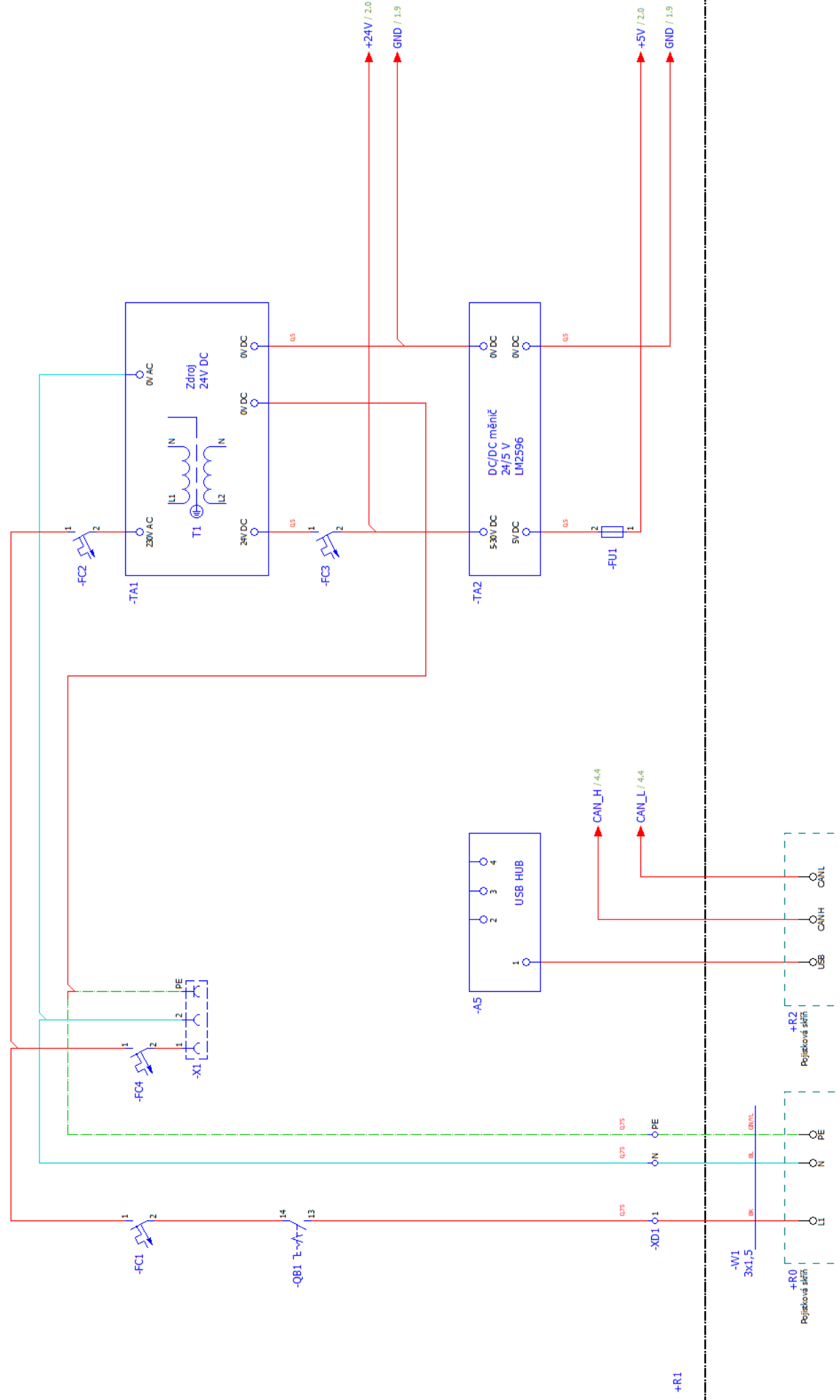
Size: A4	Date: 2020-05-01
----------	------------------

Id: 1/1	
KiCad E.D.A. pcbnew (5.0.2)-1	

## B Dokumentace Eplan

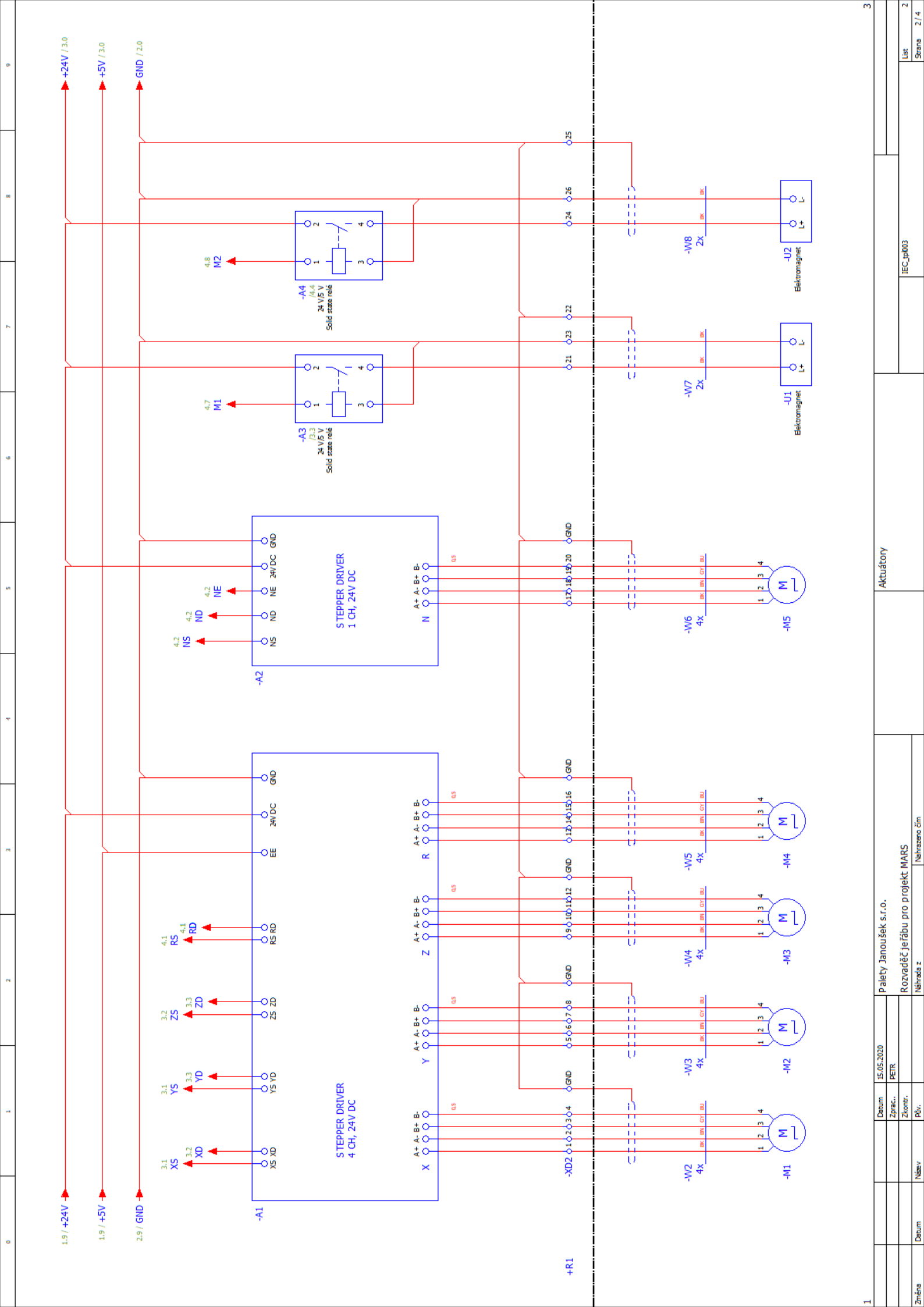
V této příloze jsou uvedeny výkresy zapojení rozvaděče. Na následujících stranách jsou uvedeny v tomto pořadí:

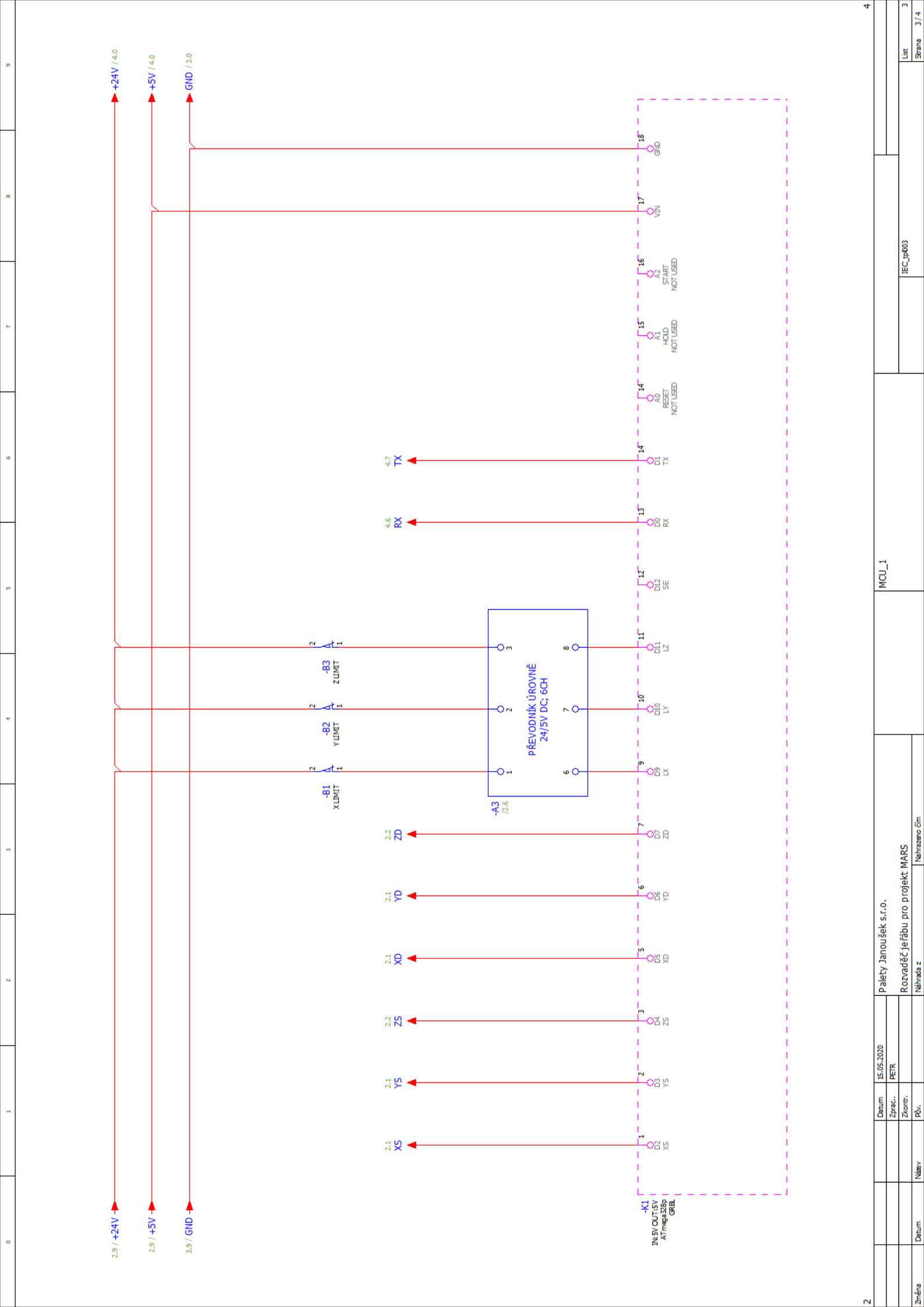
- Výkres strana 1, Napájení rozvaděče
- Výkres strana 2, Aktuátory
- Výkres strana 3, zapojení MCU Atmega328p
- Výkres strana 4, zapojení MCU Atmega2560



400V  
50Hz  
SiT'









## C Uživatelská rozhraní

V této příloze jsou uvedeny snímky uživatelských rozhraní ve čitelném rozlišení. Na následujících stranách jsou uvedeny v tomto pořadí:

- Starší rozhraní C# (Windows Forms Application)
- Novější rozhraní C# (Windows Forms Application)
- Uživatelské rozhraní LabView

Odeslaný text	Přijatý text	Text
<input type="text"/>		
Připojení	Ovladač	
COM2	VYHLEDAT	PŘIPOJIT

#### LIS

MANUÁL	NASTAVENÍ	INFORMACE
<div>Magnety</div> <div>MAGNETY</div> <div>Výhazovače</div> <div>VYHAZOVAČE</div> <div>Brzda plechu nahoru</div> <div>PŘIDRŽ PLECH</div> <div>Pojezd</div> <div>ODESLAT</div> <div>Homing</div> <div>HOMING POJEZD</div> <div>Pozice</div> <div>Nevybráno</div> <div>Zámek formy</div> <div>ZÁMEK FORMY</div> <div>Zámek kopyta</div> <div>ZÁMEK KOPYTA</div> <div>HOMING VSEHO</div> <div>Zdvih</div> <div>ODESLAT</div> <div>HOMING ZDVIH</div> <div>Výběr kopyta</div> <div>Nevybráno</div> <div>VYPNI ALARM</div>		

#### ROVNAČKY

MANUÁL	NASTAVENÍ	INFORMACE
<div>Rovnačka - malá</div> <div>VLEVO</div> <div>VPRAVO</div> <div>KE SPINAČI</div> <div>Rovnačka - velká</div> <div>Pojezd</div> <div>LEVO</div> <div>PRAVO</div> <div>ABSOLUTNĚ</div> <div>HOMING PLECH</div>		

Výrob nohu	Výrob dno
VYROB NOHU	1
POKRAČUJ	STOP
Počítadlo: NOHY: 0 DNA: 0 BOK: 0 PALETY: 0	

#### PLASMA

MANUÁL	NASTAVENÍ	INFORMACE
<div>Pohyb:</div> <div>Y+</div> <div>Z+</div> <div>100</div> <div>50</div> <div>10</div> <div>1</div> <div>X-</div> <div>X+</div> <div>Z-</div> <div>Y-</div> <div>PŘERUŠ</div> <div>DOKONČI</div> <div>FCE 50</div> <div>FCE 51</div> <div>Home po zapnutí</div> <div>HOMING</div> <div>Brzda pro řezání</div> <div>PŘIDRŽ PLECH</div> <div>Píst o 45°</div> <div>NÁKLON</div> <div>Vypnout/Zapnout</div> <div>VENTILATOR</div> <div>G-kody</div> <div>VYPAL NOHU</div> <div>Nevybráno</div> <div>Vrát' se na nulu</div> <div>RETURN</div> <div>Uřež plech</div> <div>UŘEŽ 1100</div> <div>UŘEŽ 750</div> <div>UŘEŽ DNO</div> <div>Uřeže plech nohy</div> <div>ROVNÝ ŘEZ</div>		

#### JEŘÁB

MANUÁL	NASTAVENÍ	INFORMACE
<div>Pohyb:</div> <div>X+</div> <div>Z-</div> <div>500</div> <div>100</div> <div>50</div> <div>10</div> <div>1</div> <div>Y+</div> <div>Y-</div> <div>Z+</div> <div>X-</div> <div>ROTACE</div> <div>MAGNET LEVÝ</div> <div>MAGNET PRAVÝ</div> <div>NÁKLON</div> <div>Homing</div> <div>HOMING</div> <div>Pro osu rotace</div> <div>R → HOMING</div> <div>Odvez</div> <div>NOHU</div> <div>Zadrž " ! "</div> <div>PŘERUŠ</div> <div>Pokračuj " ~ "</div> <div>DOKONČI</div> <div>BOČNICI</div> <div>DNO</div>		

**Ovládání**

### AUTOMATICKÁ VÝROBA

**Vyrob NOHU**

VYROB NOHU 0

HOTOVÝCH 0

☐ Bez odvezení ☐ S odvezením

**Vyrob DNO**

VYROB DNO 0

HOTOVÝCH 0

**Vyrob BOK**

VYROB BOK 0

HOTOVÝCH 0

☐ 750 mm ☐ 1100 mm

**Vyrob TROJCI**

VYROB TROJCI 0

HOTOVÝCH 0

POKRAČUJ

NEVYRÁBĚT ZBYLÉ DÍLY

**Počet kusů různých typů dílu:**

Typ dílu	Počet kusů
Bedny	5
Trojice	10
Dna	30
Boky	10
Nohy	20

**Celkový počet kusů za den:**

Datum	Celkový počet kusů
29.12.1899	0
30.12.1899	1
31.12.1899	10
01.01.1900	12
02.01.1900	15
03.01.1900	16
04.01.1900	17
05.01.1900	18

Status: Připojte se

AUTOMATIKA

ZOBRAZIT VŠE

NAMÁČENÍ

SVAŘOVÁNÍ

JEŘÁB

ROVNAČKA

LIS

PLASMA

OVLADAČE

**Ovládání**

### JEŘÁB - PODROBNÉ NASTAVENÍ

**Ovládání plasmy**

X+

Y+

X-

Z+

Y-

Z-

R

500  
100  
50  
10  
1

MAGNET LEVÝ

MAGNET PRAVÝ

NÁKLON

PŘERUŠ

DOKONČÍ

**Nastavení lisu**

- ☒ Vzdálenost oblouku bočnice
- ☒ Vzdálenost oblouku dlouhé bočnice
- ☒ Výška hořáku
- ☒ G55 / Počátek pálení nohy

**HOMING**

Odvez

NOHU

BOČNÍCI

DNO

**HOMING ROTACE**

Hotových

HOTOVÝCH: 1

HOTOVÝCH: 1

HOTOVÝCH: 1

**HOMING ZDVIHU**

SET ZERO

VYPNI ALARM

**Informace**

Pozice

☐ Homing

☐ Parking

☐ Unknown

Stav

☐ Idle

☐ Run

☐ Alarm

Výpis

Status: Připojte se

AUTOMATIKA

ZOBRAZIT VŠE

NAMÁČENÍ

SVAŘOVÁNÍ

JEŘÁB

ROVNAČKA

LIS

PLASMA

OVLADAČE

Baudrate

COM ports

Search ports

Connect

9600

COM2

Search

Connect

Outcoming data

Stop

2,14,0,0

Stop

LIS

PLASMA

Manual control

Settings

Magnets

Ejectors

Motor off

Hold sheet

Shape lock

Mold lock

X Axis

0

Send

Y Axis

0

Send

Shape

Choose...

Position

Choose...

Homing

Manual control

Settings

XY Axis

X+

Y+

X-

Y-

Z Axis

Z+

Z-

Distance

100

50

10

1

Homing

Nozzle

Hold

Tilt

Ventilation

Create

Divide

Cut leg

Cut long

Cut short

Cut bottom

BENDER

CRANE

Manual control

Settings

Narrow bender

Left

Right

Homing

Wide bender

0

Left

Right

Move absolute

Reset zero

Homing

Manual control

Settings

XY Axis

X+

Y+

X-

Y-

Z Axis

Z+

Z-

Distance

100

50

10

1

N Axis

N+

N-

R Axis

0

Send

Homings

Homing R

Homing XYZ

Create

Magnets

Take leg

Take bottom

Take side

Left

Right